Contribution of meat-free days, meat-free meals, and portion sizes to declines in meat consumption in the UK

author of this RMD: AV

2023-07-17

SET-UP

Load libraries and datasets

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(srvyr)
```

```
##
## Attaching package: 'srvyr'
##
## The following object is masked from 'package:stats':
##
##     filter
```

```r
library(survey)
```

```
## Loading required package: grid
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
##
## Loading required package: survival
##
## Attaching package: 'survey'
##
## The following object is masked from 'package:graphics':
##
##     dotchart
```

```r
library(effects)
```

```
## Loading required package: carData
## lattice theme set by effectsTheme()
## See ?effectsTheme for details.
```

```r
library(ggplot2)
library(RColorBrewer)
library(scales)
```

```
##
## Attaching package: 'scales'
```

```
##
## The following object is masked from 'package:purrr':
##
##     discard
##
## The following object is masked from 'package:readr':
##
##     col_factor
```

```r
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
##
##     combine
```

```r
library(cowplot)
```

```
##
## Attaching package: 'cowplot'
##
## The following object is masked from 'package:lubridate':
##
##     stamp
```

```r
library(MASS)
```

```
##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:srvyr':
##
##     select
##
## The following object is masked from 'package:dplyr':
##
##     select
```

```r
library(purrr)
library(extrafont)
```

```
## Registering fonts with R
```

```r
font_import()
```

```
## Importing fonts may take a few minutes, depending on the number of fonts and the speed of the system
## Continue? [y/n]
```

```
## Exiting.
```

```r
#set wd
setwd("/Users/alexandervonderschmidt/Library/CloudStorage/OneDrive-SharedLibraries-UniversityofEdinburgh
#upload datasets
dat <- read.csv('omega.csv')
```

DATA PREP

Setting variables and weighting structures for analysis

```
##########################EDITS AND CREATIONS######################
#remove participants with only 3 diary days (removed n = 323; n = 15,332)
dat <- dat[!(dat$DiaryDaysCompleted == 3),]

#define age brackets; <10 = 1; 11-17 = 2; 18-40 = 3; 41-59 = 4; >= 60 = 5
dat <- dat %>%
  mutate(AgeG = case_when(
    Age <= 10 ~ 1,
    Age >= 11 & Age <= 17 ~ 2,
    Age >= 18 & Age <= 40 ~ 3,
    Age >= 41 & Age <= 59 ~ 4,
    Age >= 60 ~ 5,
    TRUE ~ 99
  ))
#set reference group for age (18-40)
dat$AgeG <- as.factor(dat$AgeG)
dat$AgeG <- relevel(dat$AgeG, ref = 3)
#eqv as factor
dat$eqv <- as.factor(dat$eqv)
#sex as factor
dat$Sex <- as.factor(dat$Sex)
dat$Sex <- factor(dat$Sex, levels = c(1, 2), labels = c("M", "F"))#had some problems, just makes things
#for SMT analyses -> set 0 values to NA (to only capture meals in which meat was consumed)
varnames <- c("BsumMeatg", "BsumProcessedg", "BsumRedg", "BsumWhiteg",
              "LsumMeatg", "LsumProcessedg", "LsumRedg", "LsumWhiteg",
              "DsumMeatg", "DsumProcessedg", "DsumRedg", "DsumWhiteg")
#replace 0 with NA for each variable
for (var in varnames) {
  dat[[var]] <- ifelse(dat[[var]] == 0, NA, dat[[var]])
}

#set survey designs
#make survey year factor or numeric, depending on analyses intended to be completed
#make sure to re-set the survey design after you've changed surveyyear's variable type
dat$SurveyYear <- as.factor(dat$SurveyYear)#run for regression analyses
dat$SurveyYear <- as.numeric(dat$SurveyYear)#run for plots and also for calculating p value trends acro

#RERUN SURVEY DESIGN IF YOU'VE CHANGE ANY VARIABLES TO/FROM FACTOR/NUMERIC
#specify survey weighting structure for GLM
dat$fpc <- 15332
dat.design <-
  svydesign(
    id = ~area,
    strata = ~astrata5,
    data = dat,
    weights = ~wti,
    fpc = ~fpc
  )

#create the survey design object (for descriptive)
```

```
survey_design <- svydesign(id = ~area, strata = ~astrata5, weights = ~wti, data = dat)
```

TABLE 1

Demographic analysis

```
#specify survey weighting structure for descriptive analysis
survey_design <- dat %>%
  as_survey_design(ids = area, # cluster ids
                   weights = wti, # weight variable created above
                   strata = astrata5 # sampling was stratified by district
  )

#count age groups
#unweighted Ns
table(dat$AgeG)
```

```
##
##    3    1    2    4    5
## 2967 4295 2870 2775 2425
```

```
#weighted%s
survey_design %>%
  group_by(AgeG) %>%
  summarise(pct = survey_mean())
```

```
## # A tibble: 5 x 3
##   AgeG     pct  pct_se
##   <fct>  <dbl>   <dbl>
## 1 3      0.306  0.00627
## 2 1      0.114  0.00214
## 3 2      0.0857 0.00204
## 4 4      0.262  0.00536
## 5 5      0.232  0.00576
```

```
#sex
#unweighted Ns
table(dat$Sex)
```

```
##
##    M    F
## 7072 8260
```

```
#weighted %s
survey_design %>%
  group_by(Sex) %>%
  summarise(pct = survey_mean())
```

```
## # A tibble: 2 x 3
##   Sex     pct  pct_se
##   <fct> <dbl>   <dbl>
## 1 M     0.493 0.00619
## 2 F     0.507 0.00619
```

```
#income tertiles
#unweighted Ns
table(dat$eqv)
```

```
##
```

```
##    1    2    3
## 4449 4457 4474
```

```r
#missing
15332-(4449+4457+4474)
```

```
## [1] 1952
```

```r
#percentages of income tertiles
survey_design %>%
  group_by(eqv) %>%
  summarise(pct = survey_mean())
```

```
## # A tibble: 4 x 3
##   eqv     pct  pct_se
##   <fct> <dbl>   <dbl>
## 1 1     0.258 0.00622
## 2 2     0.282 0.00605
## 3 3     0.312 0.00676
## 4 <NA>  0.148 0.00601
```

```r
#count % of meat consumers
survey_design <- mutate(survey_design, meat_gt_0 = as.numeric(sumMeatg > 0))
table(survey_design$variables$meat_gt_0)
```

```
##
##     0     1
##   629 14703
```

```r
survey_design %>%
  group_by(meat_gt_0) %>%
  summarise(pct = survey_mean()) #95.0%
```

```
## # A tibble: 2 x 3
##   meat_gt_0    pct  pct_se
##       <dbl>  <dbl>   <dbl>
## 1         0 0.0498 0.00284
## 2         1 0.950  0.00284
```

TABLE 2

Change in meat consumption behaviours by meat category

```r
#set weighting structure for regression analysis
#set survey year to factor for regression analyses
dat$SurveyYear <- as.factor(dat$SurveyYear)
dat$fpc <- 15332
dat.design <-
  svydesign(
    id = ~area,
    strata = ~astrata5,
    data = dat,
    weights = ~wti,
    fpc = ~fpc
  )

#define custom summary function for exponentiated coefficients
#calculates 95% confidence interval values
exp_summary <- function(response_var, design) {
  #define the model formula dynamically
  model_formula <- as.formula(paste(response_var, "~ SurveyYear"))

  #fit the model
  model <- svyglm(model_formula, family=poisson(link = "log"), design = design)

  #exponentiate the sum between the intercept and SurveyYearX coefficients
  exp_diff <- exp(coef(model)["(Intercept)"] + coef(model)[-1])

  #exponentiate the confidence intervals
  conf_int <- confint(model)
  exp_conf_int <- exp(conf_int)

  #calculate the confidence intervals for the differences
  exp_diff_conf_int <- exp(conf_int["(Intercept)",] + conf_int[-1,])

  #create a new "summary" object
  exp_summary_obj <- summary(model)

  #calculate the z-values and p-values
  z_values <- coef(model) / exp_summary_obj$coefficients[, "Std. Error"]
  p_values <- 2 * pnorm(-abs(z_values))

  #replace the coefficients and confidence intervals with the exponentiated values
  exp_summary_obj$coefficients <- rbind(c(exp(coef(model)["(Intercept)"]), exp_summary_obj$coefficients
                                      cbind(exp_diff,
                                            exp_summary_obj$coefficients[-1, "Std. Error"],
                                            exp_diff_conf_int,
                                            p_values[-1]))

  #update the column names
  colnames(exp_summary_obj$coefficients) <- c("Exp(Coef)", "Std. Error", "2.5 %", "97.5 %", "Pr(>|z|)")

  #include the significance stars
```

```
  signif.stars <- options("show.signif.stars")
  if (is.logical(signif.stars) && signif.stars) {
    exp_summary_obj$coefficients <- cbind(exp_summary_obj$coefficients,
                                          exp_summary_obj$coefficients[, "Pr(>|z|)"])
    colnames(exp_summary_obj$coefficients)[ncol(exp_summary_obj$coefficients)] <- " "
    exp_summary_obj$coefficients[, " "] <- symnum(exp_summary_obj$coefficients[, "Pr(>|z|)"],
                                                  corr = FALSE, na = FALSE,
                                                  cutpoints = c(0, 0.001, 0.01, 0.05, 0.1, 1),
                                                  symbols = c("***", "**", "*", ".", " "))
  }
  #ADD A LITTLE 'difference of years 1 to 11 + 95%CI' ROW AT THE BOTTOM OF THE OUTPUT
  #correcting the name for Intercept term [it's a wee bit messed up]
  rownames(exp_summary_obj$coefficients)[rownames(exp_summary_obj$coefficients) == ""] <- "Intercept"

  #calculate the difference of the beta coefficients for Intercept and SurveyYear11
  diff_coefs <- exp_summary_obj$coefficients["Intercept", "Exp(Coef)"] - exp_summary_obj$coefficients["S

  #calculate the standard error of the difference
  se_diff <- sqrt(sum(exp_summary_obj$coefficients[c("Intercept", "SurveyYear11"), "Std. Error"]^2))

  #calculate the confidence interval for the difference
  ci_diff <- c(diff_coefs - 1.96 * se_diff, diff_coefs + 1.96 * se_diff)

  #add these values to the summary object
  exp_summary_obj$coefficients <- rbind(exp_summary_obj$coefficients,
                                        c(diff_coefs, se_diff, ci_diff[1], ci_diff[2], NA))
  rownames(exp_summary_obj$coefficients)[nrow(exp_summary_obj$coefficients)] <- "Diff"

  #round the coefficients and the confidence intervals to 2 decimal places
  exp_summary_obj$coefficients <- round(exp_summary_obj$coefficients, 2)

  return(exp_summary_obj)
}
lm_summary <- function(response_var, design) {
  #define the model formula dynamically
  model_formula <- as.formula(paste(response_var, "~ SurveyYear"))

  #fit the model
  model <- svyglm(model_formula, design = design)

  #calculate the sum between the intercept and SurveyYearX coefficients
  diff <- coef(model)["(Intercept)"] + coef(model)[-1]

  #calculate the confidence intervals
  conf_int <- confint(model)

  #calculate the confidence intervals for the differences
  diff_conf_int <- conf_int["(Intercept)",] + conf_int[-1,]

  #create a new "summary" object
  summary_obj <- summary(model)

  #calculate the t-values and p-values
```

```
  t_values <- coef(model) / summary_obj$coefficients[, "Std. Error"]
  p_values <- 2 * pt(-abs(t_values), df.residual(model))

  #replace the coefficients and confidence intervals with the calculated values
  summary_obj$coefficients <- rbind(c(coef(model)["(Intercept)"], summary_obj$coefficients[1, "Std. Erro
                                    cbind(diff,
                                          summary_obj$coefficients[-1, "Std. Error"],
                                          diff_conf_int,
                                          p_values[-1]))

  #update column names
  colnames(summary_obj$coefficients) <- c("Coef", "Std. Error", "2.5 %", "97.5 %", "Pr(>|t|)")

  #include the significance stars
  signif.stars <- options("show.signif.stars")
  if (is.logical(signif.stars) && signif.stars) {
    summary_obj$coefficients <- cbind(summary_obj$coefficients,
                                      summary_obj$coefficients[, "Pr(>|t|)"])
    colnames(summary_obj$coefficients)[ncol(summary_obj$coefficients)] <- " "
    summary_obj$coefficients[, " "] <- symnum(summary_obj$coefficients[, "Pr(>|t|)"],
                                              corr = FALSE, na = FALSE,
                                              cutpoints = c(0, 0.001, 0.01, 0.05, 0.1, 1),
                                              symbols = c("***", "**", "*", ".", " "))
  }

  # ADD A LITTLE 'difference of years 1 to 11 + 95%CI' ROW AT THE BOTTOM OF THE OUTPUT
  #correct the name for Intercept term [it's a wee bit messed up]
  rownames(summary_obj$coefficients)[rownames(summary_obj$coefficients) == ""] <- "Intercept"

  #calculate the difference of the intercept + SurveyYear11 coefficients
  diff_coefs <- summary_obj$coefficients["Intercept", "Coef"] - summary_obj$coefficients["SurveyYear11"

  #calculate the standard error of the difference
  se_diff <- sqrt(sum(summary_obj$coefficients[c("Intercept", "SurveyYear11"), "Std. Error"]^2))

  #calculate the confidence interval for the difference
  ci_diff <- c(diff_coefs - 1.96 * se_diff, diff_coefs + 1.96 * se_diff)

  #add these values to the summary at the bottom
  summary_obj$coefficients <- rbind(summary_obj$coefficients,
                                    c(diff_coefs, se_diff, ci_diff[1], ci_diff[2], NA))
  rownames(summary_obj$coefficients)[nrow(summary_obj$coefficients)] <- "Diff"

  #round the coefficients and the confidence intervals to 2 decimal places
  summary_obj$coefficients <- round(summary_obj$coefficients, 2)

  return(summary_obj)
}

##MEAT DAYS##
exp_summary(response_var = "MeatDays", design = dat.design)
```

```
##
## Call:
```

```
## svyglm(formula = model_formula, design = design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Exp(Coef) Std. Error 2.5 % 97.5 % Pr(>|z|)
## Intercept         3.27       0.01  3.20   3.34   <2e-16 ***
## SurveyYear2       3.19       0.02  3.02   3.23     0.17
## SurveyYear3       3.17       0.02  3.12   3.36     0.08 .
## SurveyYear4       3.14       0.02  2.97   3.17     0.02 *
## SurveyYear5       3.26       0.02  3.22   3.44     0.81
## SurveyYear6       3.16       0.02  2.98   3.20     0.06 .
## SurveyYear7       3.19       0.02  3.14   3.39     0.19
## SurveyYear8       3.19       0.02  3.00   3.24     0.21
## SurveyYear9       3.09       0.02  3.01   3.31     0.02 *
## SurveyYear10      3.08       0.02  2.90   3.13   <2e-16 ***
## SurveyYear11      3.03       0.02  2.97   3.22   <2e-16 ***
## Diff              0.24       0.02  0.20   0.29       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3821775)
##
## Number of Fisher Scoring iterations: 4
```

```r
exp_summary(response_var = "ProcessedDays", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Exp(Coef) Std. Error 2.5 % 97.5 % Pr(>|z|)
## Intercept         1.77       0.03  1.68   1.87   <2e-16 ***
## SurveyYear2       1.76       0.04  1.54   1.81     0.93
## SurveyYear3       1.65       0.04  1.60   1.89     0.10 .
## SurveyYear4       1.72       0.04  1.51   1.75     0.45
## SurveyYear5       1.75       0.04  1.70   2.00     0.78
## SurveyYear6       1.75       0.04  1.53   1.80     0.79
## SurveyYear7       1.73       0.04  1.67   1.99     0.61
## SurveyYear8       1.63       0.05  1.41   1.69     0.07 .
## SurveyYear9       1.59       0.05  1.52   1.84     0.03 *
## SurveyYear10      1.62       0.05  1.40   1.68     0.06 .
## SurveyYear11      1.59       0.04  1.54   1.83     0.02 *
## Diff              0.18       0.05  0.07   0.28       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.9531377)
```

```
##
## Number of Fisher Scoring iterations: 5
```

```r
exp_summary(response_var = "RedDays", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                Exp(Coef) Std. Error 2.5 % 97.5 % Pr(>|z|)
## Intercept           1.56       0.02  1.49   1.63   <2e-16 ***
## SurveyYear2         1.46       0.04  1.30   1.50     0.08 .
## SurveyYear3         1.35       0.04  1.31   1.53   <2e-16 ***
## SurveyYear4         1.43       0.04  1.27   1.46     0.01 **
## SurveyYear5         1.51       0.04  1.46   1.72     0.44
## SurveyYear6         1.40       0.04  1.24   1.45     0.01 **
## SurveyYear7         1.37       0.04  1.32   1.57   <2e-16 ***
## SurveyYear8         1.39       0.04  1.23   1.43   <2e-16 ***
## SurveyYear9         1.34       0.05  1.28   1.54   <2e-16 ***
## SurveyYear10        1.29       0.04  1.14   1.34   <2e-16 ***
## SurveyYear11        1.23       0.05  1.18   1.41   <2e-16 ***
## Diff                0.33       0.05  0.23   0.43       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.8855853)
##
## Number of Fisher Scoring iterations: 5
```

```r
exp_summary(response_var = "WhiteDays", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                Exp(Coef) Std. Error 2.5 % 97.5 % Pr(>|z|)
## Intercept           1.42       0.03  1.35   1.50   <2e-16 ***
## SurveyYear2         1.40       0.04  1.22   1.44     0.71
## SurveyYear3         1.46       0.04  1.42   1.68     0.47
## SurveyYear4         1.48       0.04  1.30   1.51     0.34
## SurveyYear5         1.47       0.04  1.44   1.67     0.36
## SurveyYear6         1.43       0.04  1.25   1.47     0.88
## SurveyYear7         1.59       0.04  1.55   1.81   <2e-16 ***
## SurveyYear8         1.63       0.04  1.43   1.68   <2e-16 ***
## SurveyYear9         1.57       0.04  1.51   1.80     0.02 *
```

```
## SurveyYear10      1.62       0.04  1.42   1.67   <2e-16 ***
## SurveyYear11      1.57       0.04  1.52   1.79     0.02 *
## Diff             -0.15       0.05 -0.24  -0.05       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.8850367)
##
## Number of Fisher Scoring iterations: 5
```

```r
exp_summary(response_var = "NoMeatDays", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Exp(Coef) Std. Error 2.5 % 97.5 % Pr(>|z|)
## Intercept         0.73       0.05  0.66   0.81   <2e-16 ***
## SurveyYear2       0.81       0.07  0.63   0.84     0.17
## SurveyYear3       0.83       0.07  0.80   1.07     0.08 .
## SurveyYear4       0.86       0.07  0.68   0.89     0.02 *
## SurveyYear5       0.74       0.08  0.71   0.95     0.81
## SurveyYear6       0.84       0.07  0.66   0.88     0.05 *
## SurveyYear7       0.81       0.08  0.77   1.05     0.19
## SurveyYear8       0.81       0.08  0.62   0.86     0.21
## SurveyYear9       0.91       0.09  0.85   1.20     0.01 **
## SurveyYear10      0.92       0.07  0.72   0.96   <2e-16 ***
## SurveyYear11      0.97       0.07  0.93   1.24   <2e-16 ***
## Diff             -0.24       0.09 -0.42  -0.07       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1.446405)
##
## Number of Fisher Scoring iterations: 5
```

```r
##Meat occasions##
exp_summary(response_var = "avgMeatokaj", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Exp(Coef) Std. Error 2.5 % 97.5 % Pr(>|z|)
## Intercept         1.24       0.02  1.20   1.28   <2e-16 ***
```

```
## SurveyYear2       1.20       0.03  1.10   1.22     0.22
## SurveyYear3       1.17       0.03  1.15   1.27     0.02 *
## SurveyYear4       1.19       0.02  1.10   1.21     0.13
## SurveyYear5       1.22       0.02  1.20   1.33     0.58
## SurveyYear6       1.18       0.03  1.09   1.20     0.09 .
## SurveyYear7       1.21       0.03  1.19   1.33     0.49
## SurveyYear8       1.21       0.03  1.11   1.23     0.39
## SurveyYear9       1.17       0.03  1.13   1.29     0.07 .
## SurveyYear10      1.15       0.03  1.05   1.17     0.01 **
## SurveyYear11      1.13       0.03  1.11   1.24    <2e-16 ***
## Diff              0.11       0.03  0.04   0.17       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.2845649)
##
## Number of Fisher Scoring iterations: 4
```

```
exp_summary(response_var = "avgProcessedokaj", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Exp(Coef) Std. Error 2.5 % 97.5 % Pr(>|z|)
## Intercept         0.54       0.03  0.51   0.57    <2e-16 ***
## SurveyYear2       0.54       0.05  0.46   0.56     1.00
## SurveyYear3       0.52       0.05  0.50   0.61     0.38
## SurveyYear4       0.52       0.04  0.45   0.53     0.28
## SurveyYear5       0.54       0.05  0.52   0.62     0.85
## SurveyYear6       0.54       0.05  0.46   0.56     0.88
## SurveyYear7       0.54       0.05  0.52   0.64     0.93
## SurveyYear8       0.50       0.05  0.42   0.52     0.10 .
## SurveyYear9       0.48       0.05  0.46   0.57     0.05 *
## SurveyYear10      0.49       0.05  0.41   0.51     0.04 *
## SurveyYear11      0.50       0.05  0.48   0.58     0.09 .
## Diff              0.04       0.06 -0.07   0.16       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3807208)
##
## Number of Fisher Scoring iterations: 5
```

```
exp_summary(response_var = "avgRedokaj", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design, family = poisson(link = "log"))
##
```

```
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Exp(Coef) Std. Error 2.5 % 97.5 % Pr(>|z|)
## Intercept         0.44       0.03  0.41   0.47   <2e-16 ***
## SurveyYear2       0.40       0.04  0.35   0.41     0.05 *
## SurveyYear3       0.37       0.04  0.36   0.42   <2e-16 ***
## SurveyYear4       0.40       0.04  0.35   0.41     0.04 *
## SurveyYear5       0.42       0.05  0.41   0.49     0.41
## SurveyYear6       0.39       0.05  0.33   0.40     0.01 **
## SurveyYear7       0.39       0.05  0.37   0.46     0.02 *
## SurveyYear8       0.39       0.05  0.34   0.40     0.01 **
## SurveyYear9       0.38       0.05  0.36   0.44   <2e-16 ***
## SurveyYear10      0.35       0.05  0.30   0.37   <2e-16 ***
## SurveyYear11      0.33       0.05  0.32   0.39   <2e-16 ***
## Diff              0.11       0.06 -0.01   0.22       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.2973773)
##
## Number of Fisher Scoring iterations: 5
```

```
exp_summary(response_var = "avgWhiteokaj", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Exp(Coef) Std. Error 2.5 % 97.5 % Pr(>|z|)
## Intercept         0.40       0.03  0.38   0.43   <2e-16 ***
## SurveyYear2       0.40       0.05  0.34   0.42     0.81
## SurveyYear3       0.41       0.05  0.40   0.48     0.66
## SurveyYear4       0.43       0.05  0.37   0.44     0.25
## SurveyYear5       0.42       0.04  0.41   0.48     0.44
## SurveyYear6       0.40       0.05  0.34   0.41     0.87
## SurveyYear7       0.46       0.05  0.44   0.53     0.01 **
## SurveyYear8       0.48       0.05  0.41   0.50   <2e-16 ***
## SurveyYear9       0.46       0.05  0.44   0.54     0.02 *
## SurveyYear10      0.46       0.05  0.39   0.47     0.01 **
## SurveyYear11      0.45       0.05  0.44   0.53     0.02 *
## Diff             -0.05       0.06 -0.16   0.07       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3297555)
##
## Number of Fisher Scoring iterations: 5
```

```r
##g per occasion##
lm_summary(response_var = "gperokajMeat", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                Coef Std. Error 2.5 % 97.5 % Pr(>|t|)
## Intercept     85.76       1.85 82.13  89.38   <2e-16 ***
## SurveyYear2   81.96       2.37 73.69  82.98     0.11
## SurveyYear3   85.81       2.41 84.70  94.17     0.98
## SurveyYear4   79.97       2.37 71.70  80.99     0.01 **
## SurveyYear5   78.91       2.29 78.04  87.03   <2e-16 ***
## SurveyYear6   81.25       2.35 73.02  82.22     0.05 *
## SurveyYear7   82.26       2.51 80.96  90.82     0.16
## SurveyYear8   79.30       2.34 71.09  80.27     0.01 **
## SurveyYear9   80.93       2.37 79.90  89.21     0.04 *
## SurveyYear10 77.27       2.41 68.92  78.36   <2e-16 ***
## SurveyYear11 76.12       2.57 74.71  84.78   <2e-16 ***
## Diff           9.64       3.16  3.43  15.84       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1357.625)
##
## Number of Fisher Scoring iterations: 2
```

```r
lm_summary(response_var = "gperokajProcessed", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                Coef Std. Error 2.5 % 97.5 % Pr(>|t|)
## Intercept     63.21       2.27 58.77  67.66   <2e-16 ***
## SurveyYear2   60.71       2.77 50.84  61.69     0.37
## SurveyYear3   63.72       3.02 62.25  74.08     0.87
## SurveyYear4   61.17       2.85 51.12  62.32     0.47
## SurveyYear5   58.74       3.01 57.30  69.09     0.14
## SurveyYear6   59.73       2.85 49.69  60.88     0.22
## SurveyYear7   55.50       2.85 54.36  65.55     0.01 **
## SurveyYear8   55.38       2.90 45.25  56.61     0.01 **
## SurveyYear9   53.01       2.80 51.97  62.94   <2e-16 ***
## SurveyYear10 51.09       2.64 41.46  51.82   <2e-16 ***
```

```
## SurveyYear11 52.64       2.72 51.76  62.42   <2e-16 ***
## Diff         10.57       3.54  3.63  17.51       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1457.199)
##
## Number of Fisher Scoring iterations: 2
```

```r
lm_summary(response_var = "gperokajRed", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##               Coef Std. Error 2.5 % 97.5 % Pr(>|t|)
## Intercept    89.73       3.17 83.51  95.95   <2e-16 ***
## SurveyYear2  77.20       3.98 63.18  78.79   <2e-16 ***
## SurveyYear3  86.06       4.12 84.19 100.35     0.37
## SurveyYear4  79.15       4.30 64.49  81.37     0.01 **
## SurveyYear5  76.63       3.77 75.46  90.23   <2e-16 ***
## SurveyYear6  74.98       4.10 60.72  76.80   <2e-16 ***
## SurveyYear7  78.73       4.21 76.69  93.21     0.01 **
## SurveyYear8  73.58       3.79 59.93  74.79   <2e-16 ***
## SurveyYear9  78.29       4.35 75.96  93.04     0.01 **
## SurveyYear10 73.51       4.05 59.35  75.23   <2e-16 ***
## SurveyYear11 69.69       3.99 68.07  83.73   <2e-16 ***
## Diff         20.04       5.10 10.05  30.04       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2991.584)
##
## Number of Fisher Scoring iterations: 2
```

```r
lm_summary(response_var = "gperokajWhite", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##               Coef Std. Error 2.5 % 97.5 % Pr(>|t|)
## Intercept    84.65       2.62 79.51  89.79   <2e-16 ***
## SurveyYear2  83.04       3.48 71.07  84.72     0.64
## SurveyYear3  86.85       3.74 84.66  99.33     0.56
```

```
## SurveyYear4  83.04       3.25 71.53  84.27     0.62
## SurveyYear5  81.27       3.34 79.86  92.95     0.31
## SurveyYear6  82.16       3.55 70.06  83.97     0.48
## SurveyYear7  88.58       4.04 85.80 101.65     0.33
## SurveyYear8  83.00       3.23 71.52  84.18     0.61
## SurveyYear9  84.43       3.24 83.21  95.93     0.95
## SurveyYear10 80.14       3.49 68.16  81.84     0.20
## SurveyYear11 79.74       3.64 77.75  92.02     0.18
## Diff          4.91       4.48 -3.88  13.70       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2540.63)
##
## Number of Fisher Scoring iterations: 2
```

```
lm_summary(response_var = "okajTotalGrams", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
## Intercept    468.85      11.15 446.99 490.71   <2e-16 ***
## SurveyYear2  449.85      15.61 397.37 458.61     0.22
## SurveyYear3  443.00      17.25 431.02 498.70     0.13
## SurveyYear4  475.51      15.86 422.55 484.75     0.67
## SurveyYear5  464.59      15.17 456.70 516.19     0.78
## SurveyYear6  492.19      16.35 438.26 502.41     0.15
## SurveyYear7  479.73      15.38 471.41 531.76     0.48
## SurveyYear8  478.44      15.56 426.05 487.11     0.54
## SurveyYear9  503.56      16.99 492.10 558.74     0.04 *
## SurveyYear10 512.63      17.12 457.18 524.35     0.01 **
## SurveyYear11 516.60      20.55 498.15 578.76     0.02 *
## Diff         -47.75      23.38 -93.57  -1.93       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 86586.73)
##
## Number of Fisher Scoring iterations: 2
```

```
###CALCULATING P-VALUES###
#set survey year to numeric for p-values
dat$SurveyYear <- as.numeric(dat$SurveyYear)
dat$fpc <- 15332
dat.design <-
  svydesign(
    id = ~area,
    strata = ~astrata5,
```

```r
    data = dat,
    weights = ~wti,
    fpc = ~fpc
  )

#p values (use only after setting SurveyYear to numeric)
summary(svyglm(MeatDays ~ SurveyYear, family=poisson(link = "log"), dat.design))
```

```
##
## Call:
## svyglm(formula = MeatDays ~ SurveyYear, design = dat.design,
##     family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.180703   0.008476 139.307  < 2e-16 ***
## SurveyYear  -0.005072   0.001408  -3.601 0.000326 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3824452)
##
## Number of Fisher Scoring iterations: 4
```

```r
summary(svyglm(ProcessedDays ~ SurveyYear, family=poisson(link = "log"), dat.design))
```

```
##
## Call:
## svyglm(formula = ProcessedDays ~ SurveyYear, design = dat.design,
##     family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.578832   0.019064  30.362  < 2e-16 ***
## SurveyYear  -0.009482   0.003070  -3.088  0.00205 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.9533318)
##
## Number of Fisher Scoring iterations: 5
```

```r
summary(svyglm(RedDays ~ SurveyYear, family=poisson(link = "log"), dat.design))
```

```
##
## Call:
## svyglm(formula = RedDays ~ SurveyYear, design = dat.design, family = poisson(link = "log"))
```

```
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.427971   0.018416  23.239  < 2e-16 ***
## SurveyYear  -0.016249   0.003003  -5.411 7.13e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.8870363)
##
## Number of Fisher Scoring iterations: 5
```

```r
summary(svyglm(WhiteDays ~ SurveyYear, family=poisson(link = "log"), dat.design))
```

```
##
## Call:
## svyglm(formula = WhiteDays ~ SurveyYear, design = dat.design,
##     family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.326987   0.019548  16.727  < 2e-16 ***
## SurveyYear  0.014505   0.002941   4.932 8.89e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.8862231)
##
## Number of Fisher Scoring iterations: 5
```

```r
summary(svyglm(NoMeatDays ~ SurveyYear, family=poisson(link = "log"), dat.design))
```

```
##
## Call:
## svyglm(formula = NoMeatDays ~ SurveyYear, design = dat.design,
##     family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.290494   0.034030  -8.536  < 2e-16 ***
## SurveyYear   0.019156   0.005261   3.641 0.000279 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for poisson family taken to be 1.448813)
##
## Number of Fisher Scoring iterations: 5
```

```r
summary(svyglm(avgMeatokaj ~ SurveyYear, family=poisson(link = "log"), dat.design))
```

```
##
## Call:
## svyglm(formula = avgMeatokaj ~ SurveyYear, design = dat.design,
##     family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.202899   0.012296  16.501   <2e-16 ***
## SurveyYear  -0.005037   0.001972  -2.554   0.0107 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.2848375)
##
## Number of Fisher Scoring iterations: 4
```

```r
summary(svyglm(avgProcessedokaj ~ SurveyYear, family=poisson(link = "log"), dat.design))
```

```
##
## Call:
## svyglm(formula = avgProcessedokaj ~ SurveyYear, design = dat.design,
##     family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.603435   0.022729 -26.550  < 2e-16 ***
## SurveyYear  -0.009386   0.003558  -2.638  0.00841 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3812651)
##
## Number of Fisher Scoring iterations: 5
```

```r
summary(svyglm(avgRedokaj ~ SurveyYear, family=poisson(link = "log"), dat.design))
```

```
##
## Call:
## svyglm(formula = avgRedokaj ~ SurveyYear, design = dat.design,
##     family = poisson(link = "log"))
##
```

```
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.84642    0.02088 -40.545  < 2e-16 ***
## SurveyYear  -0.01702    0.00333  -5.111 3.55e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.2986477)
##
## Number of Fisher Scoring iterations: 5
```

```r
summary(svyglm(avgWhiteokaj ~ SurveyYear, family=poisson(link = "log"), dat.design))
```

```
##
## Call:
## svyglm(formula = avgWhiteokaj ~ SurveyYear, design = dat.design,
##     family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.932965   0.022813 -40.896  < 2e-16 ***
## SurveyYear   0.016252   0.003422   4.749 2.21e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3305674)
##
## Number of Fisher Scoring iterations: 5
```

```r
summary(svyglm(gperokajMeat ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajMeat ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  85.2089     1.0768  79.133  < 2e-16 ***
## SurveyYear   -0.7251     0.1598  -4.537 6.09e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1361.076)
```

```
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(gperokajProcessed ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajProcessed ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  65.1316     1.2687  51.338  < 2e-16 ***
## SurveyYear   -1.2388     0.1763  -7.025 3.06e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1459.032)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(gperokajRed ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajRed ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  86.2503     1.8448  46.753  < 2e-16 ***
## SurveyYear   -1.3797     0.2669  -5.169 2.62e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 3004.704)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(gperokajWhite ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajWhite ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  85.2273     1.5906  53.583   <2e-16 ***
## SurveyYear   -0.3063     0.2318  -1.321    0.187
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2545.826)
##
## Number of Fisher Scoring iterations: 2
```

```r
#count #of participants in each survey year
table(dat$SurveyYear)
```

```
##
##    1    2    3    4    5    6    7    8    9   10   11
## 1629 1639 1529 1902 1169 1331 1328 1339 1216 1174 1076
```

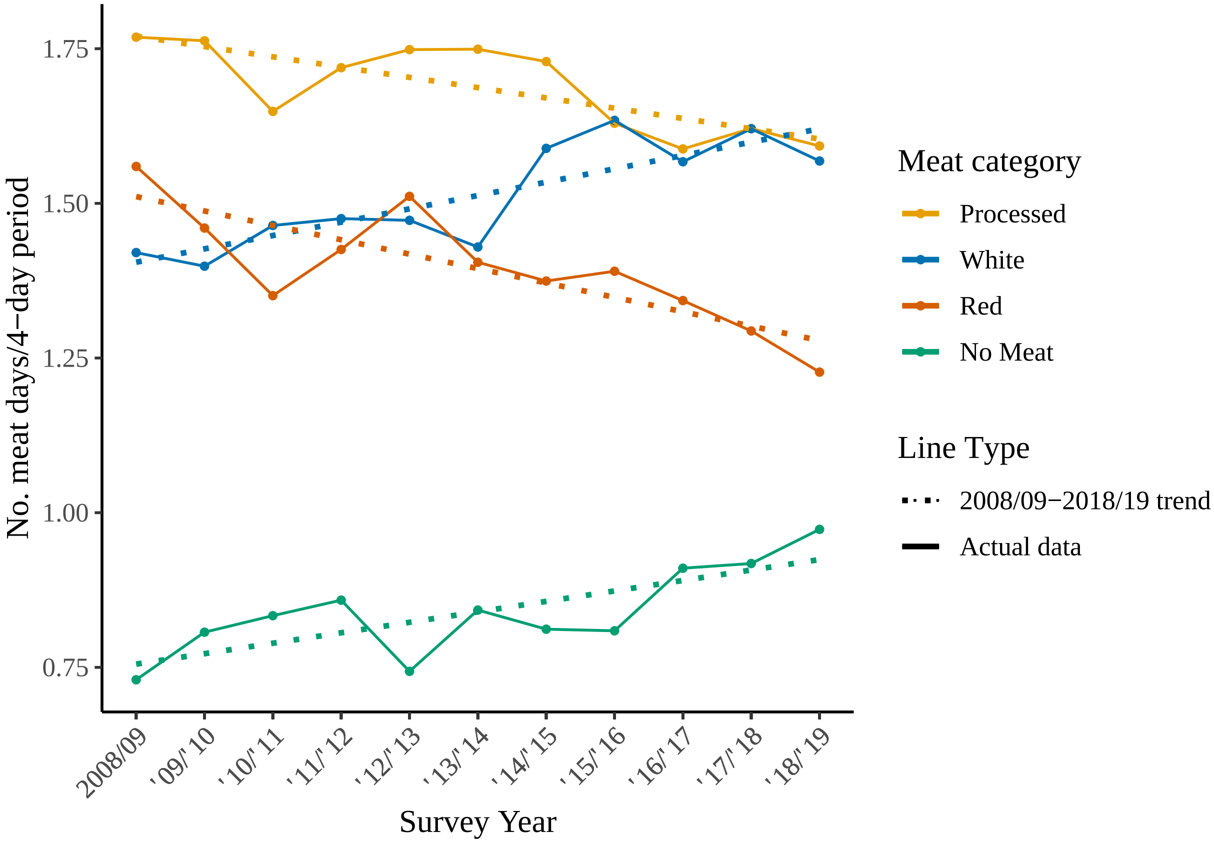
FIGURE 1

Trends of meat-eating behaviours by year

```
#Set survey year factor for plots w/linear regression
dat$SurveyYear <- as.factor(dat$SurveyYear)
dat$fpc <- 15332
dat.design <-
  svydesign(
    id = ~area,
    strata = ~astrata5,
    data = dat,
    weights = ~wti,
    fpc = ~fpc
  )

#function for Survey Year x axis labels
custom_x_labels <- function(x) {
  labels <- ifelse(x == 1, "2008/09", sprintf("'%02d/'%02d", x + 7, (x + 7) %% 100 + 1))
  return(labels)
}
#line plot of meat trends
#Days
m2 <- svyglm(ProcessedDays ~ SurveyYear, family=poisson(link = "log"), dat.design)
m3 <- svyglm(RedDays ~ SurveyYear, family=poisson(link = "log"), dat.design)
m4 <- svyglm(WhiteDays ~ SurveyYear, family=poisson(link = "log"), dat.design)
m5 <- svyglm(NoMeatDays ~ SurveyYear, family=poisson(link = "log"), dat.design)
#fitted values for each model
survey_years <- unique(dat.design$variables$SurveyYear)
predictions <- data.frame(
  SurveyYear = rep(survey_years, 4),
  Category = factor(rep(c("ProcessedDays", "RedDays", "WhiteDays", "NoMeatDays"), each = length(survey_
  PredictedDays = c(predict(m2, newdata = data.frame(SurveyYear = survey_years), type = "response"),
                    predict(m3, newdata = data.frame(SurveyYear = survey_years), type = "response"),
                    predict(m4, newdata = data.frame(SurveyYear = survey_years), type = "response"),
                    predict(m5, newdata = data.frame(SurveyYear = survey_years), type = "response"))
)
#create a custom color palette using colorblind friendly colors
color_palette <- c("#E69F00", "#0072B2", "#D55E00", "#009E73") #order: processed (orange), white (blue)
#correct order
predictions$Category <- factor(predictions$Category, levels = c("ProcessedDays", "WhiteDays", "RedDays"
#category names
levels(predictions$Category) <- c("Processed", "White", "Red", "No Meat")
#convert SurveyYear to numeric
predictions$SurveyYear <- as.numeric(as.character(predictions$SurveyYear))
#create plot
plot1 <- ggplot(predictions, aes(x = SurveyYear, y = PredictedDays, color = Category, group = Category)
  geom_point(size = 1) +
  geom_line(aes(linetype = "solid")) +
  geom_smooth(method = "glm", formula = 'y ~ x', se = FALSE, aes(linetype = "dotted", group = Category)
  scale_color_manual(values = color_palette) +
  scale_linetype_manual(name = "Line Type",
                        values = c("solid" = "solid", "dotted" = "dotted"),
                        labels = c("solid" = "Actual data", "dotted" = "2008/09-2018/19 trend")) +
```

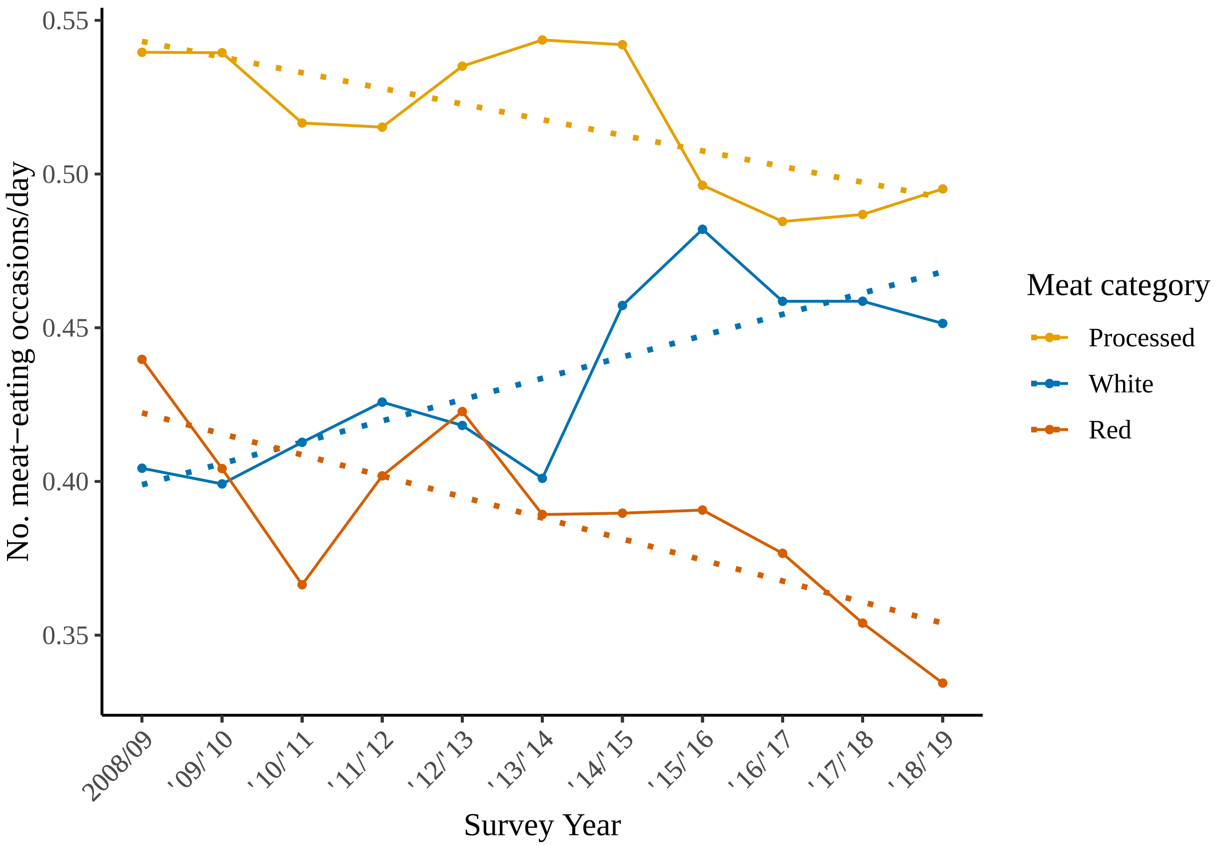
```
  labs(x = "Survey Year", y = "No. meat days/4-day period", color = "Meat category") +
  scale_x_continuous(breaks = predictions$SurveyYear, labels = custom_x_labels) +
  theme_classic() +
  theme(text = element_text(family = "Times New Roman", size = 12),
        axis.text.x = element_text(angle = 45, hjust = 1)) +
  guides(linetype = guide_legend(override.aes = list(color = "black")))
print(plot1)
```




```
#Occasions
m2 <- svyglm(avgProcessedokaj ~ SurveyYear, family=poisson(link = "log"), dat.design)
m3 <- svyglm(avgRedokaj ~ SurveyYear, family=poisson(link = "log"), dat.design)
m4 <- svyglm(avgWhiteokaj ~ SurveyYear, family=poisson(link = "log"), dat.design)
#fitted values for each model
survey_years <- unique(dat.design$variables$SurveyYear)
predictions <- data.frame(
  SurveyYear = rep(survey_years, 3),
  Category = factor(rep(c("avgProcessedokaj", "avgRedokaj", "avgWhiteokaj"), each = length(survey_years
  PredictedOccasions = c(predict(m2, newdata = data.frame(SurveyYear = survey_years), type = "response"
                         predict(m3, newdata = data.frame(SurveyYear = survey_years), type = "response"
                         predict(m4, newdata = data.frame(SurveyYear = survey_years), type = "response"
)
#colorblind friendly colors
color_palette <- c("#E69F00", "#0072B2", "#D55E00") #order: processed (orange), white (blue), red (red)
#correct order
predictions$Category <- factor(predictions$Category, levels = c("avgProcessedokaj", "avgWhiteokaj", "avg
#names for xlab
levels(predictions$Category) <- c("Processed", "White", "Red")
```

```
#convert SurveyYear to numeric
predictions$SurveyYear <- as.numeric(as.character(predictions$SurveyYear))
#create plot
plot2 <- ggplot(predictions, aes(x = SurveyYear, y = PredictedOccasions, color = Category, group = Categ
  geom_point(size = 1) +
  geom_line() +
  geom_smooth(method = "glm", se = FALSE, linetype = "dotted", aes(group = Category)) + #this adds the 
  scale_color_manual(values = color_palette) +
  labs(x = "Survey Year", y = "No. meat-eating occasions/day", color = "Meat category") +
  scale_x_continuous(breaks = predictions$SurveyYear, labels = custom_x_labels) +
  theme_classic() +
  theme(text = element_text(family = "Times New Roman", size = 12),
        axis.text.x = element_text(angle = 45, hjust = 1)) +
  guides(linetype = guide_legend(override.aes = list(color = "black")))
print(plot2)
```

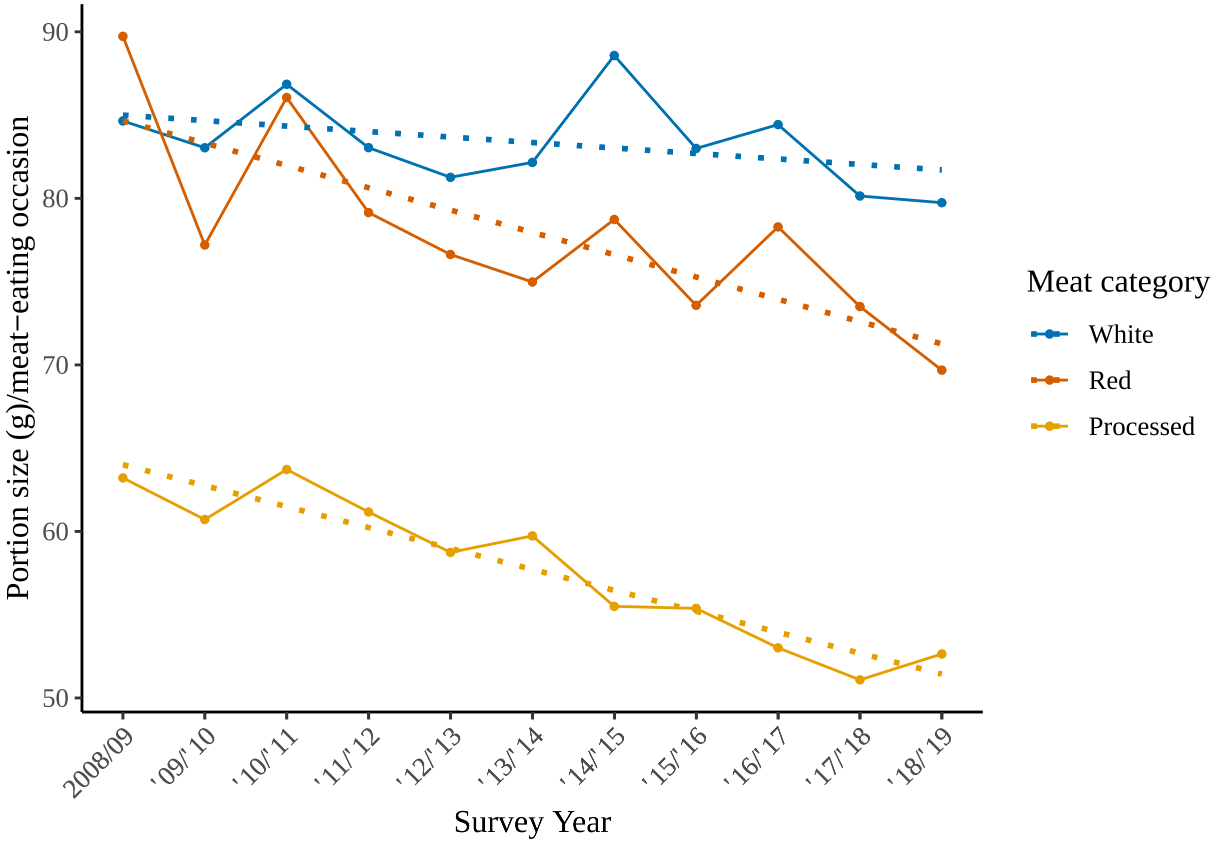
```
## `geom_smooth()` using formula = 'y ~ x'
```




```
#portion size
m2 <- svyglm(gperokajProcessed ~ SurveyYear, family=poisson(link = "log"), dat.design)
m3 <- svyglm(gperokajRed ~ SurveyYear, family=poisson(link = "log"), dat.design)
m4 <- svyglm(gperokajWhite ~ SurveyYear, family=poisson(link = "log"), dat.design)
#fitted values for each model
survey_years <- unique(dat.design$variables$SurveyYear)
predictions <- data.frame(
  SurveyYear = rep(survey_years, 3),
  Category = factor(rep(c("gperokajProcessed", "gperokajRed", "gperokajWhite"), each = length(survey_yea
```

```
  PredictedPortion = c(predict(m2, newdata = data.frame(SurveyYear = survey_years), type = "response"),
                       predict(m3, newdata = data.frame(SurveyYear = survey_years), type = "response"),
                       predict(m4, newdata = data.frame(SurveyYear = survey_years), type = "response"))
)
#colorblind friendly
color_palette <- c("#0072B2", "#D55E00", "#E69F00") #order: white (blue), red (red), processed (orange)
#correct order
predictions$Category <- factor(predictions$Category, levels = c("gperokajWhite", "gperokajRed", "gperoka
#names for xlab
levels(predictions$Category) <- c("White", "Red", "Processed")
#convert SurveyYear to numeric
predictions$SurveyYear <- as.numeric(as.character(predictions$SurveyYear))
#create plot
plot3 <- ggplot(predictions, aes(x = SurveyYear, y = PredictedPortion, color = Category, group = Catego
  geom_point(size = 1) +
  geom_line() +
  geom_smooth(method = "glm", se = FALSE, linetype = "dotted", aes(group = Category)) + #this adds the
  scale_color_manual(values = color_palette) +
  labs(x = "Survey Year", y = "Portion size (g)/meat-eating occasion", color = "Meat category") +
  scale_x_continuous(breaks = predictions$SurveyYear, labels = custom_x_labels) +
  theme_classic() +
  theme(text = element_text(family = "Times New Roman", size = 12),
        axis.text.x = element_text(angle = 45, hjust = 1)) +
  guides(linetype = guide_legend(override.aes = list(color = "black")))
print(plot3)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



```r
#combine all into 1 figure
#Remove the legend from plot2 and plot3
plot2 <- plot2 + theme(legend.position = "none")
plot3 <- plot3 + theme(legend.position = "none")
#extract the legend from plot1
legend_grob <- cowplot::get_legend(plot1)
#remove the legend from plot1
plot1 <- plot1 + theme(legend.position = "none")
#combine the plots and legend into a single plot
top_row <- cowplot::plot_grid(plot1, plot2, nrow = 1)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```r
bottom_row <- cowplot::plot_grid(plot3, legend_grob, nrow = 1, rel_widths = c(1, 1))
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```r
combined_plot <- cowplot::plot_grid(top_row, bottom_row, ncol = 1, rel_heights = c(1, 1))
print(combined_plot)
```

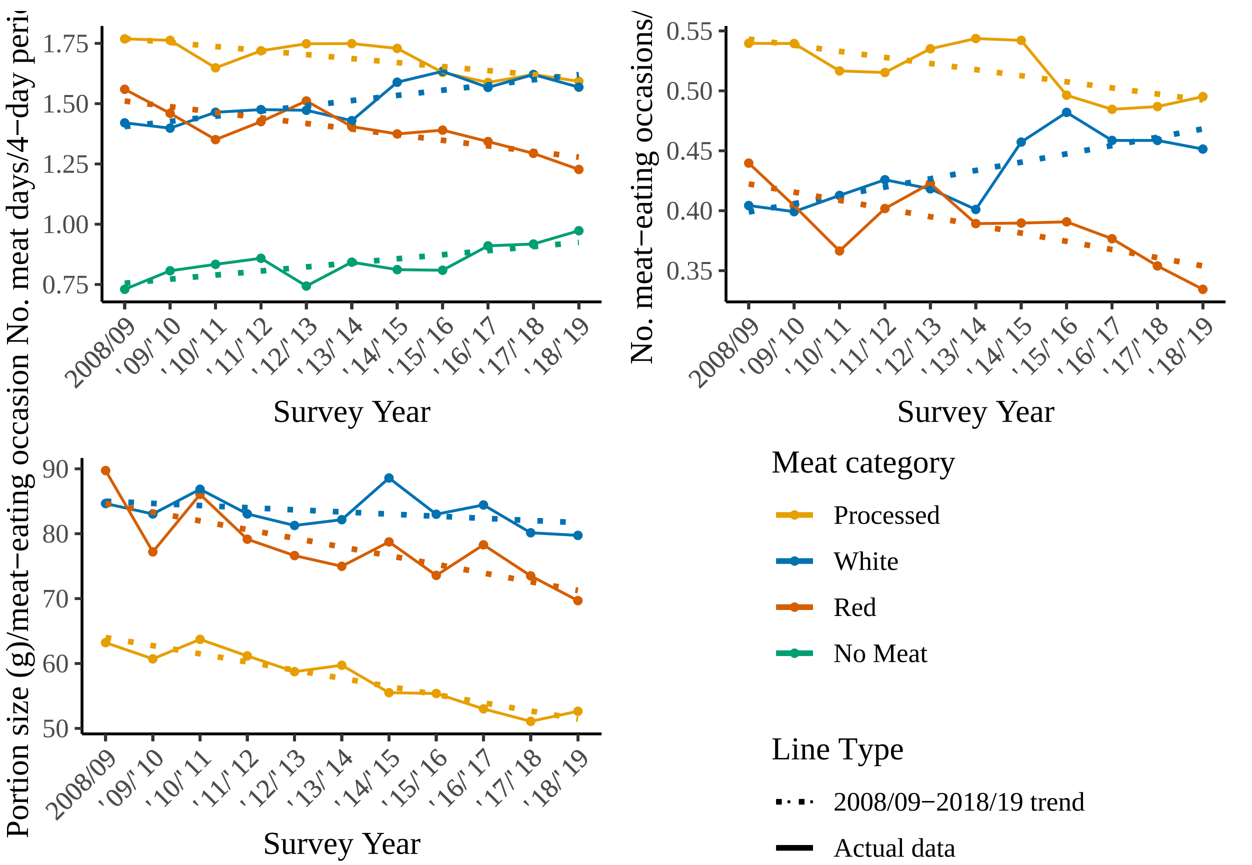


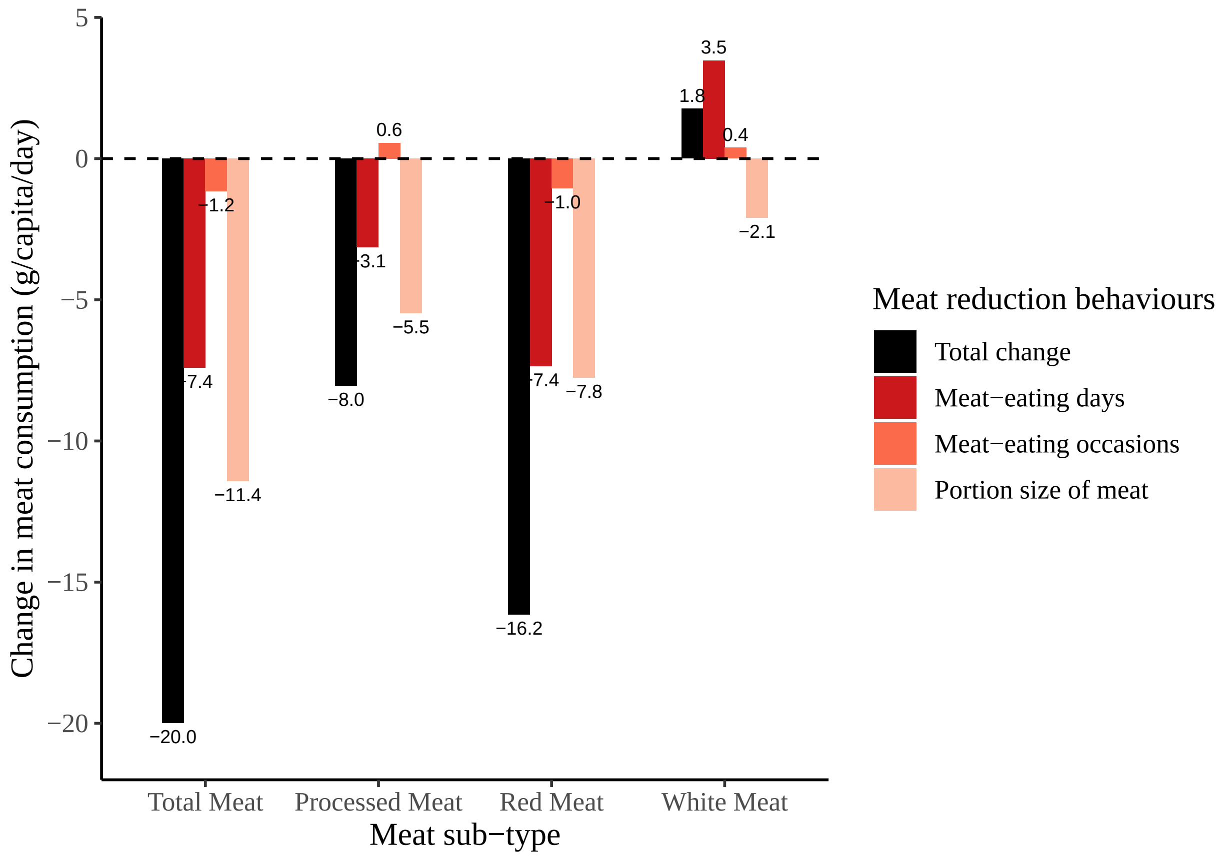
FIGURE 2

Decomposition analysis plot

```
#create dataset (values pulled from decomposition analysis section of this code)
meat_data <- data.frame(
  Meat = factor(c("Total Meat", "Processed Meat", "Red Meat", "White Meat"),
                levels = c("Total Meat", "Processed Meat", "Red Meat", "White Meat")),
  Total_Delta = c(-79.94, -32.18, -64.61, 7.08),
  Days_Delta = c(-29.61, -12.53, -29.41, 13.92),
  Occasions_Delta = c(-4.62, 2.25, -4.18, 1.55),
  Portion_Size_Delta = c(-45.71, -21.90, -31.02, -8.39)
)
meat_data_divided <- meat_data
numeric_columns <- sapply(meat_data, is.numeric)
#divide numeric columns by 4 (to represent daily intake)
meat_data_divided[, numeric_columns] <- meat_data[, numeric_columns] / 4
#rename back to original
meat_data <- meat_data_divided
print(meat_data)
```

```
##             Meat Total_Delta Days_Delta Occasions_Delta Portion_Size_Delta
## 1     Total Meat    -19.9850    -7.4025         -1.1550           -11.4275
## 2 Processed Meat     -8.0450    -3.1325          0.5625            -5.4750
## 3       Red Meat    -16.1525    -7.3525         -1.0450            -7.7550
## 4     White Meat      1.7700     3.4800          0.3875            -2.0975
```

```
#transform
melted_data <- reshape2::melt(meat_data, id.vars = "Meat")
#make plot
bar_plot <- ggplot(melted_data, aes(x = Meat, y = value, fill = variable)) +
  geom_bar(stat = "identity", position = "dodge", width = 0.5) +
  scale_fill_manual(values = c("Total_Delta" = "black", "Days_Delta" = "#CB181D", "Occasions_Delta" = "#
                    labels = c("Total_Delta" = "Total change",
                               "Days_Delta" = "Meat-eating days",
                               "Occasions_Delta" = "Meat-eating occasions",
                               "Portion_Size_Delta" = "Portion size of meat")) +
  labs(x = "Meat sub-type", y = "Change in meat consumption (g/capita/day)", fill = "Meat reduction beha
  theme_classic() +
  theme(text = element_text(family = "Times New Roman", size = 12))
#define the y-axis limits (didn't like the cuts it was giving me)
y_limits <- c(-22, 5)
#update plot with modified y-axis limits/breaks and a dashed line at y=0
bar_plot <- bar_plot +
  scale_y_continuous(limits = y_limits, expand = c(0, 0),
                     breaks = seq(-20, 5, by = 5)) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "black") +
  geom_text(aes(label = sprintf("%.1f", value), y = value, group = variable, vjust = ifelse(value >= 0,
            position = position_dodge(width = 0.5), size = 2.5)
print(bar_plot)
```



Change in meat consumption (g/capita/day)
5
0
−5
−10
−15
−20
−20.0
−7.4
−1.2
−11.4
−8.0
−3.1
0.6
−5.5
−16.2
−7.4
−1.0
−7.8
1.8
3.5
0.4
−2.1
Total Meat
Processed Meat
Red Meat
White Meat
Meat sub−type
Meat reduction behaviours
Total change
Meat−eating days
Meat−eating occasions
Portion size of meat

FIGURE 3

Meat days as a proportion of the population, by year

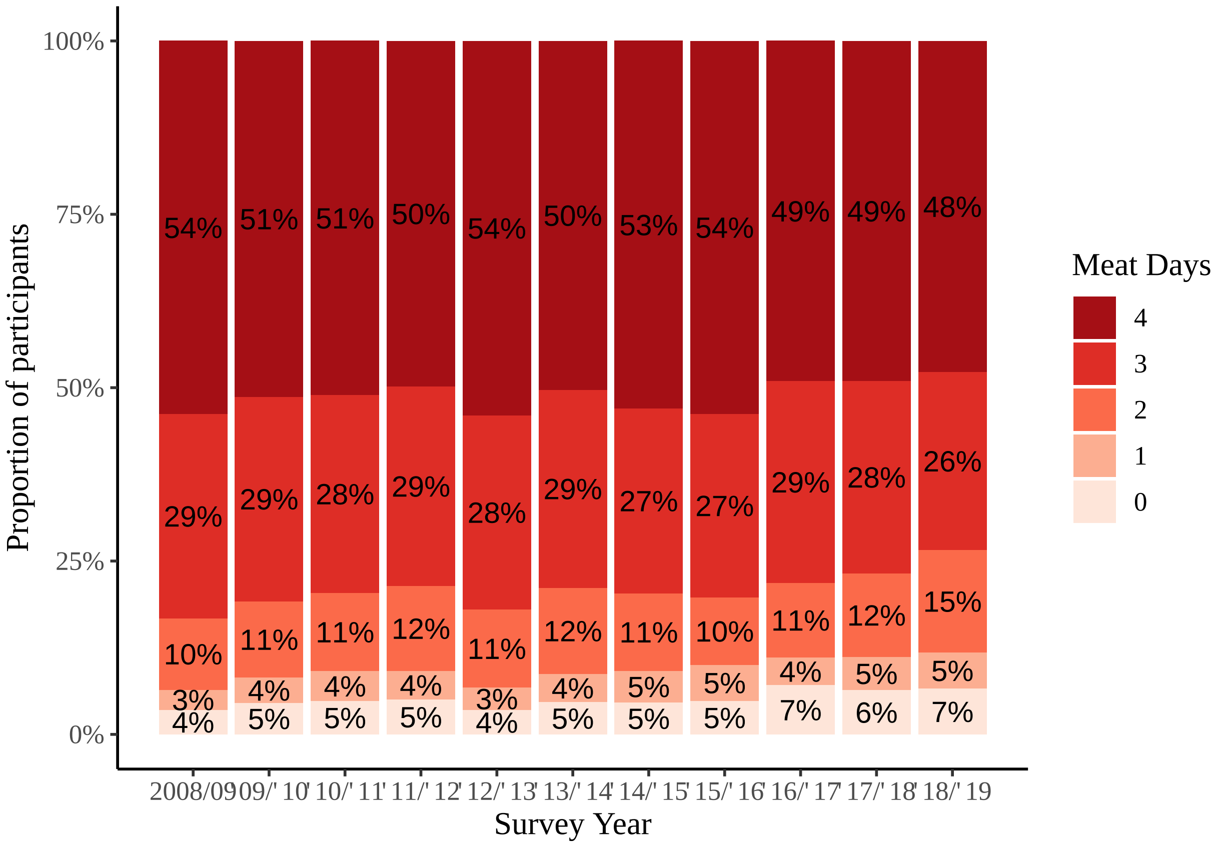
```
custom_x_labels <- function(x) {
  labels <- ifelse(x == 1, "2008/09", sprintf("'%02d/'%02d", x + 7, (x + 7) %% 100 + 1))
  return(labels)
}

#total meat days
#set the weighting structure a srvyr object with the survey design
dat_svy <- as_survey(survey_design)
#create categorical variable for each level of MeatDays by SurveyYear
meat_days_prop <- dat_svy %>%
  group_by(SurveyYear) %>%
  summarize(prop_0 = survey_mean(MeatDays == 0),
            prop_1 = survey_mean(MeatDays == 1),
            prop_2 = survey_mean(MeatDays == 2),
            prop_3 = survey_mean(MeatDays == 3),
            prop_4 = survey_mean(MeatDays == 4))
meat_days_prop
```

```
## # A tibble: 11 x 11
##    SurveyYear prop_0 prop_0_se prop_1 prop_1_se prop_2 prop_2_se prop_3
##         <dbl>  <dbl>     <dbl>  <dbl>     <dbl>  <dbl>     <dbl>  <dbl>
##  1          1 0.0357   0.00739 0.0289   0.00708 0.103     0.0114  0.294
##  2          2 0.0456   0.00889 0.0365   0.00705 0.110     0.0123  0.295
##  3          3 0.0484   0.00808 0.0429   0.00763 0.113     0.0123  0.285
##  4          4 0.0507   0.00828 0.0412   0.00686 0.122     0.0121  0.288
##  5          5 0.0351   0.00647 0.0327   0.00724 0.112     0.0128  0.280
##  6          6 0.0466   0.00888 0.0407   0.00697 0.124     0.0135  0.286
##  7          7 0.0461   0.00899 0.0456   0.00835 0.112     0.0128  0.266
##  8          8 0.0485   0.0106  0.0522   0.00888 0.0967    0.0114  0.265
##  9          9 0.0710   0.0158  0.0402   0.00827 0.107     0.0122  0.291
## 10         10 0.0640   0.00956 0.0477   0.00952 0.121     0.0122  0.277
## 11         11 0.0660   0.0112  0.0522   0.00836 0.148     0.0159  0.257
## # i 3 more variables: prop_3_se <dbl>, prop_4 <dbl>, prop_4_se <dbl>
```

```
#subset the columns that end in "_se"
se_cols <- grep("_se$", names(meat_days_prop))
#Remove the columns that end in "_se"
meat_days_prop_no_se <- meat_days_prop[, -se_cols]
#transform  data from wide to long format
meat_days_prop_long <- pivot_longer(meat_days_prop_no_se, cols = -SurveyYear, names_to = "MeatDays", va
#create stacked bar plot
plot1 <- ggplot(meat_days_prop_long, aes(x = SurveyYear, y = proportion, fill = factor(str_remove(MeatD
  geom_col() +
  scale_fill_brewer(palette = "Reds", direction = -1) +
  labs(x = "Survey Year", y = "Proportion of participants", fill = "Meat Days") +
  scale_x_continuous(breaks = meat_days_prop$SurveyYear, labels = custom_x_labels) +
  geom_text(aes(label = paste0(round(proportion*100),"%")),
            position = position_stack(vjust = 0.5)) +
  theme_classic() +
  theme(text = element_text(family = "Times New Roman", size = 12)) +
  scale_y_continuous(labels = percent, breaks = seq(0, 1, by = 0.25))
```

```
plot1
```

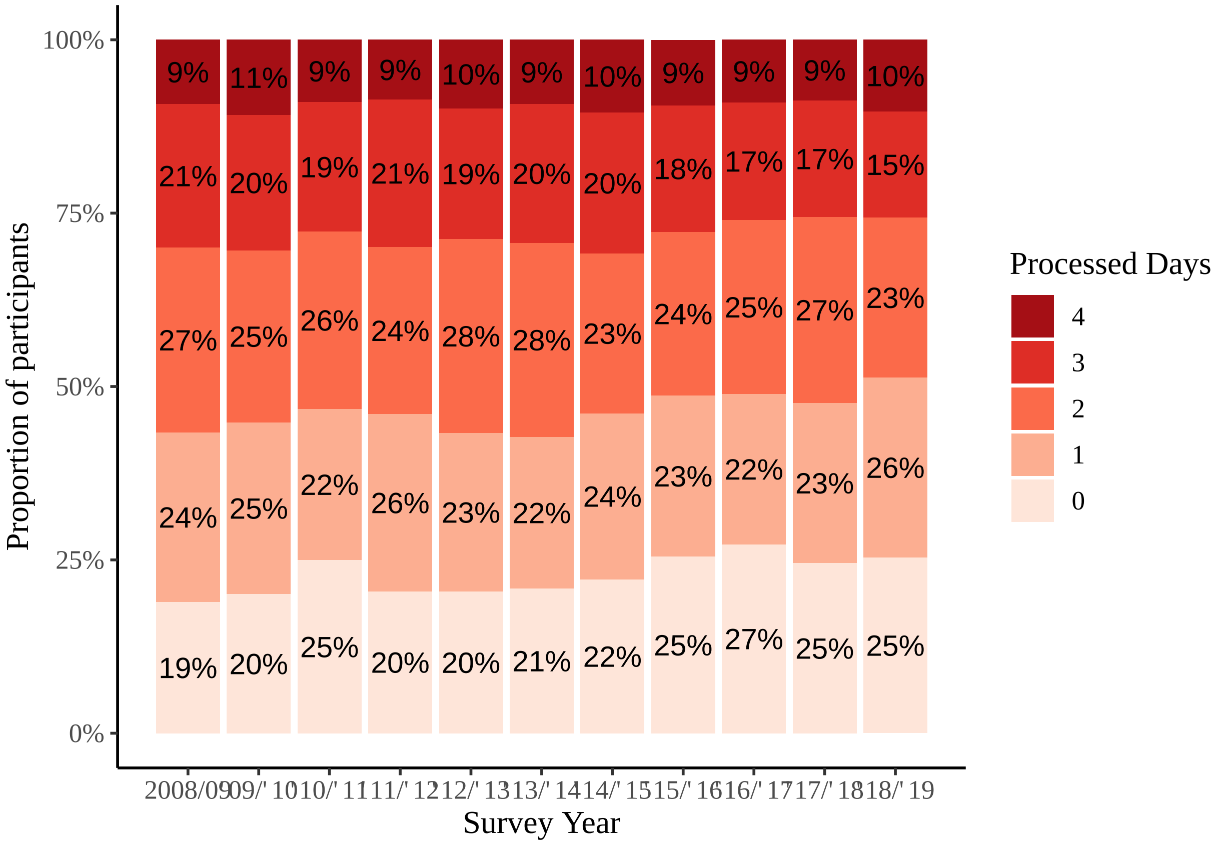



```
#processed meat days
#create categorical variable for each level of Processed by SurveyYear
Processed_days_prop <- dat_svy %>%
  group_by(SurveyYear) %>%
  summarize(prop_0 = survey_mean(ProcessedDays == 0),
            prop_1 = survey_mean(ProcessedDays == 1),
            prop_2 = survey_mean(ProcessedDays == 2),
            prop_3 = survey_mean(ProcessedDays == 3),
            prop_4 = survey_mean(ProcessedDays == 4))
Processed_days_prop
```

```
## # A tibble: 11 x 11
##    SurveyYear prop_0 prop_0_se prop_1 prop_1_se prop_2 prop_2_se prop_3
##         <dbl>  <dbl>     <dbl>  <dbl>     <dbl>  <dbl>     <dbl>  <dbl>
##  1          1  0.189    0.0134  0.244    0.0156  0.267    0.0173  0.207
##  2          2  0.201    0.0166  0.248    0.0166  0.248    0.0153  0.195
##  3          3  0.250    0.0204  0.218    0.0165  0.256    0.0155  0.187
##  4          4  0.205    0.0167  0.255    0.0168  0.241    0.0152  0.213
##  5          5  0.204    0.0172  0.229    0.0170  0.279    0.0195  0.189
##  6          6  0.209    0.0177  0.218    0.0147  0.280    0.0189  0.200
##  7          7  0.222    0.0199  0.240    0.0190  0.230    0.0163  0.203
##  8          8  0.255    0.0209  0.232    0.0152  0.236    0.0181  0.183
##  9          9  0.273    0.0221  0.216    0.0160  0.251    0.0198  0.170
## 10         10  0.245    0.0205  0.231    0.0173  0.268    0.0184  0.168
## 11         11  0.254    0.0175  0.259    0.0168  0.231    0.0175  0.153
```

```
## # i 3 more variables: prop_3_se <dbl>, prop_4 <dbl>, prop_4_se <dbl>
```

```
#subset the columns that end in "_se"
se_cols <- grep("_se$", names(Processed_days_prop))
#remove the columns that end in "_se"
Processed_days_prop_no_se <- Processed_days_prop[, -se_cols]
#transform data from wide to long format
Processed_days_prop_long <- pivot_longer(Processed_days_prop_no_se, cols = -SurveyYear, names_to = "Proc
#stacked bar plot
plot2 <- ggplot(Processed_days_prop_long, aes(x = SurveyYear, y = proportion, fill = factor(str_remove(P
  geom_col() +
  scale_fill_brewer(palette = "Reds", direction = -1) +
  labs(x = "Survey Year", y = "Proportion of participants", fill = "Processed Days") +
  scale_x_continuous(breaks = Processed_days_prop$SurveyYear, labels = custom_x_labels) +
  geom_text(aes(label = paste0(round(proportion*100),"%")),
            position = position_stack(vjust = 0.5)) +
  theme_classic() +
  theme(text = element_text(family = "Times New Roman", size = 12)) +
  scale_y_continuous(labels = percent, breaks = seq(0, 1, by = 0.25))
plot2
```

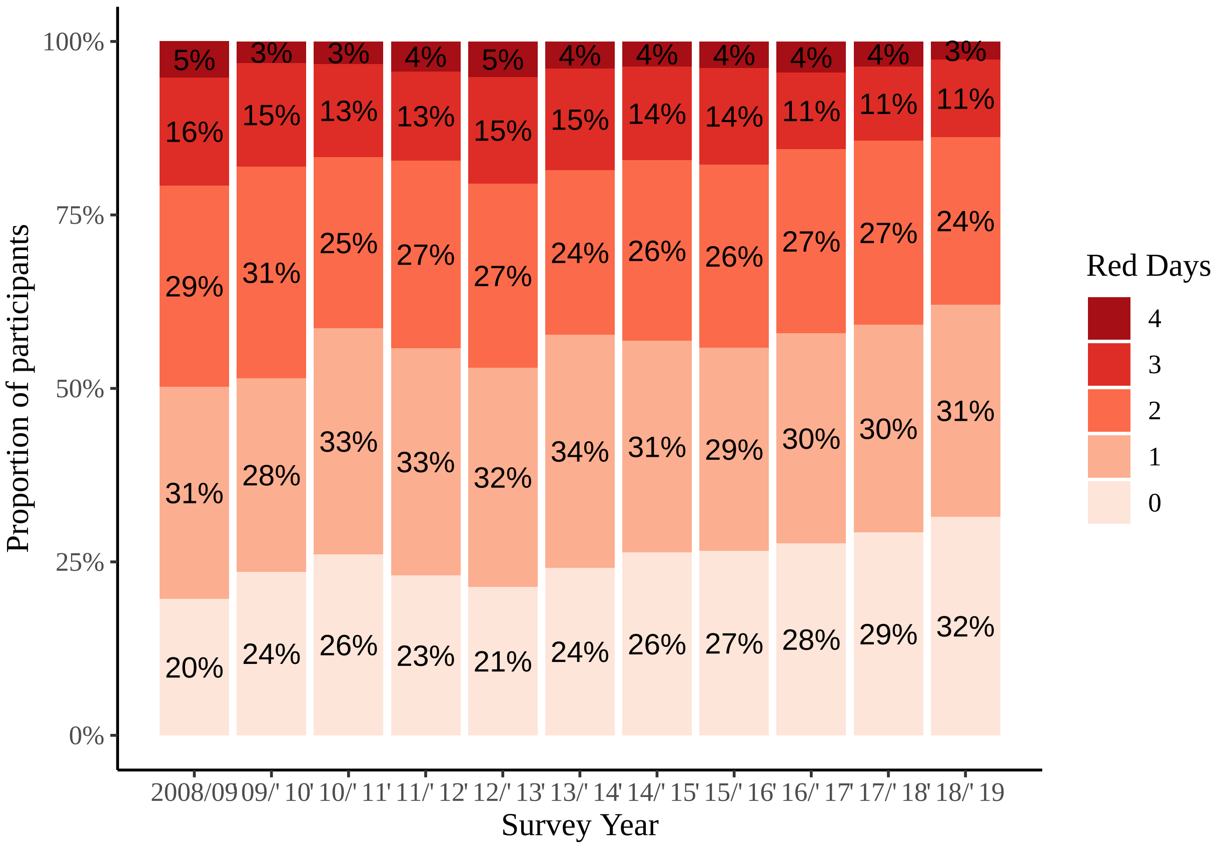



```
#red meat days
#create categorical variable for each level of RedDays by SurveyYear
Red_days_prop <- dat_svy %>%
  group_by(SurveyYear) %>%
  summarize(prop_0 = survey_mean(RedDays == 0),
            prop_1 = survey_mean(RedDays == 1),
            prop_2 = survey_mean(RedDays == 2),
```

```r
            prop_3 = survey_mean(RedDays == 3),
            prop_4 = survey_mean(RedDays == 4))
Red_days_prop
```

```
## # A tibble: 11 x 11
##    SurveyYear prop_0 prop_0_se prop_1 prop_1_se prop_2 prop_2_se prop_3
##         <dbl>  <dbl>     <dbl>  <dbl>     <dbl>  <dbl>     <dbl>  <dbl>
##  1          1  0.197    0.0140  0.306    0.0149  0.290    0.0165  0.155
##  2          2  0.236    0.0157  0.278    0.0161  0.305    0.0179  0.149
##  3          3  0.261    0.0187  0.326    0.0172  0.247    0.0162  0.134
##  4          4  0.231    0.0168  0.328    0.0202  0.270    0.0172  0.129
##  5          5  0.214    0.0169  0.316    0.0176  0.265    0.0167  0.154
##  6          6  0.242    0.0183  0.336    0.0189  0.237    0.0161  0.147
##  7          7  0.264    0.0194  0.305    0.0181  0.260    0.0178  0.135
##  8          8  0.266    0.0185  0.292    0.0176  0.264    0.0179  0.139
##  9          9  0.277    0.0210  0.302    0.0173  0.266    0.0178  0.110
## 10         10  0.293    0.0191  0.299    0.0184  0.265    0.0170  0.107
## 11         11  0.315    0.0193  0.306    0.0167  0.241    0.0183  0.112
## # i 3 more variables: prop_3_se <dbl>, prop_4 <dbl>, prop_4_se <dbl>
```

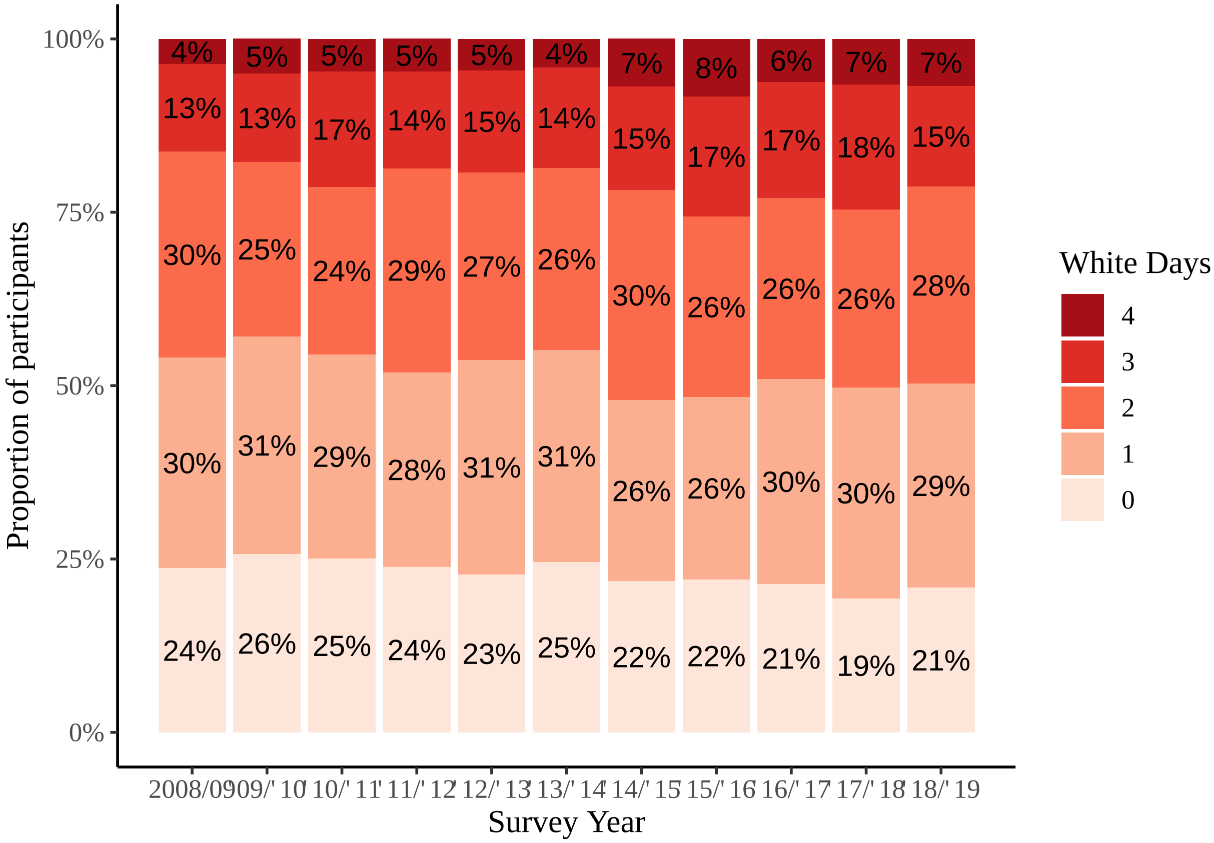
```r
#subset the columns that end in "_se"
se_cols <- grep("_se$", names(Red_days_prop))
#remove the columns that end in "_se"
Red_days_prop_no_se <- Red_days_prop[, -se_cols]
#transform the data from wide to long format
Red_days_prop_long <- pivot_longer(Red_days_prop_no_se, cols = -SurveyYear, names_to = "RedDays", values
#stacked bar plot
plot3 <- ggplot(Red_days_prop_long, aes(x = SurveyYear, y = proportion, fill = factor(str_remove(RedDays
  geom_col() +
  scale_fill_brewer(palette = "Reds", direction = -1) +
  labs(x = "Survey Year", y = "Proportion of participants", fill = "Red Days") +
  scale_x_continuous(breaks = Red_days_prop$SurveyYear, labels = custom_x_labels) +
  geom_text(aes(label = paste0(round(proportion*100),"%")),
            position = position_stack(vjust = 0.5)) +
  theme_classic() +
  theme(text = element_text(family = "Times New Roman", size = 12)) +
  scale_y_continuous(labels = percent, breaks = seq(0, 1, by = 0.25))
plot3
```




```
#white meat days
#create categorical variable for each level of WhiteDays by SurveyYear
white_days_prop <- dat_svy %>%
  group_by(SurveyYear) %>%
  summarize(prop_0 = survey_mean(WhiteDays == 0),
            prop_1 = survey_mean(WhiteDays == 1),
            prop_2 = survey_mean(WhiteDays == 2),
            prop_3 = survey_mean(WhiteDays == 3),
            prop_4 = survey_mean(WhiteDays == 4))
white_days_prop
```

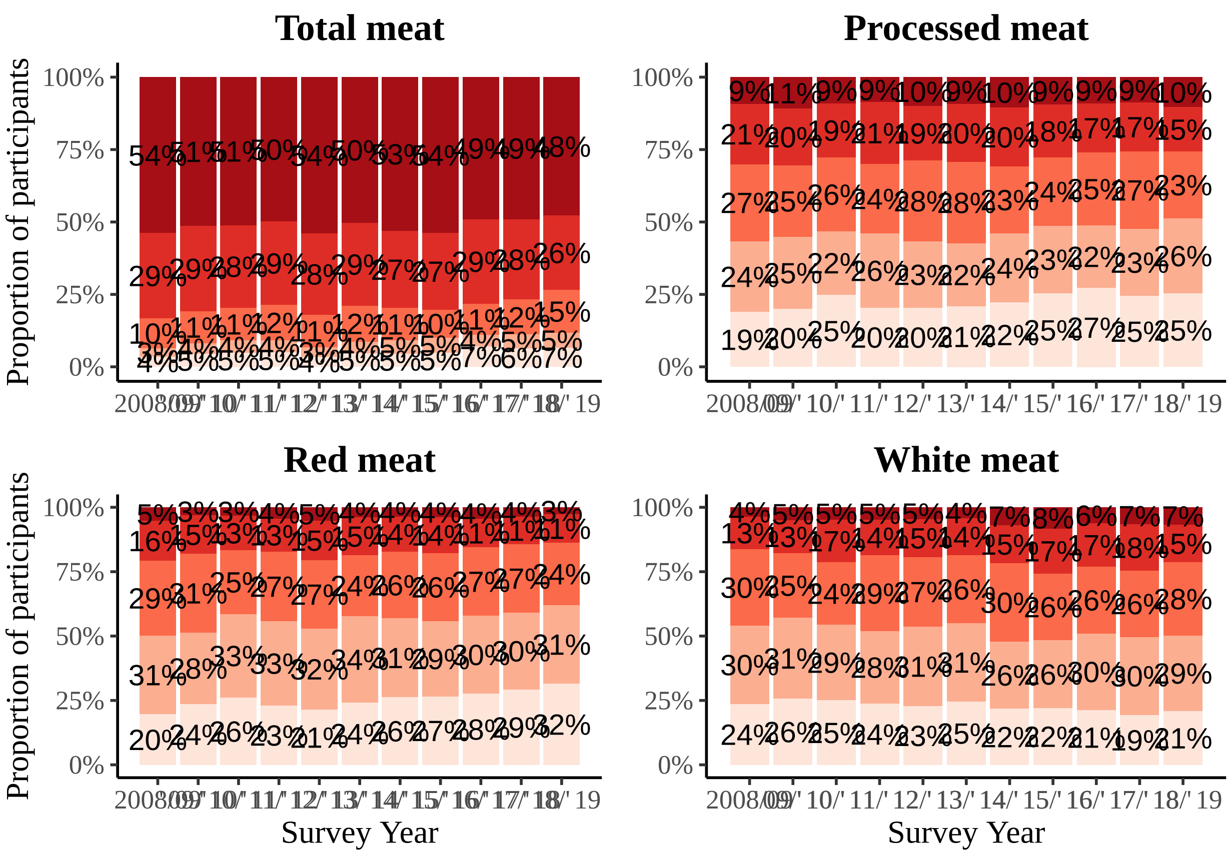
```
## # A tibble: 11 x 11
##    SurveyYear prop_0 prop_0_se prop_1 prop_1_se prop_2 prop_2_se prop_3
##         <dbl>  <dbl>     <dbl>  <dbl>     <dbl>  <dbl>     <dbl>  <dbl>
##  1          1  0.237    0.0157  0.304    0.0187  0.297    0.0170  0.127
##  2          2  0.257    0.0169  0.314    0.0165  0.252    0.0147  0.127
##  3          3  0.251    0.0150  0.295    0.0167  0.241    0.0166  0.167
##  4          4  0.239    0.0148  0.281    0.0176  0.295    0.0164  0.139
##  5          5  0.228    0.0193  0.310    0.0210  0.270    0.0177  0.148
##  6          6  0.246    0.0172  0.305    0.0166  0.263    0.0177  0.145
##  7          7  0.218    0.0159  0.261    0.0158  0.303    0.0160  0.149
##  8          8  0.221    0.0181  0.263    0.0176  0.260    0.0181  0.174
##  9          9  0.214    0.0171  0.296    0.0193  0.261    0.0163  0.168
## 10         10  0.193    0.0155  0.304    0.0189  0.257    0.0176  0.181
## 11         11  0.209    0.0167  0.294    0.0180  0.284    0.0184  0.146
## # i 3 more variables: prop_3_se <dbl>, prop_4 <dbl>, prop_4_se <dbl>
```

```r
#subset the columns that end in "_se"
se_cols <- grep("_se$", names(white_days_prop))
#remove the columns that end in "_se"
white_days_prop_no_se <- white_days_prop[, -se_cols]
#transform the data from wide to long format
white_days_prop_long <- pivot_longer(white_days_prop_no_se, cols = -SurveyYear, names_to = "WhiteDays",
#stacked bar plot
plot4 <- ggplot(white_days_prop_long, aes(x = SurveyYear, y = proportion, fill = factor(str_remove(Whit
  geom_col() +
  scale_fill_brewer(palette = "Reds", direction = -1) +
  labs(x = "Survey Year", y = "Proportion of participants", fill = "White Days") +
  scale_x_continuous(breaks = white_days_prop$SurveyYear, labels = custom_x_labels) +
  geom_text(aes(label = paste0(round(proportion*100),"%")),
            position = position_stack(vjust = 0.5)) +
  theme_classic() +
  theme(text = element_text(family = "Times New Roman", size = 12)) +
  scale_y_continuous(labels = percent, breaks = seq(0, 1, by = 0.25))
plot4
```

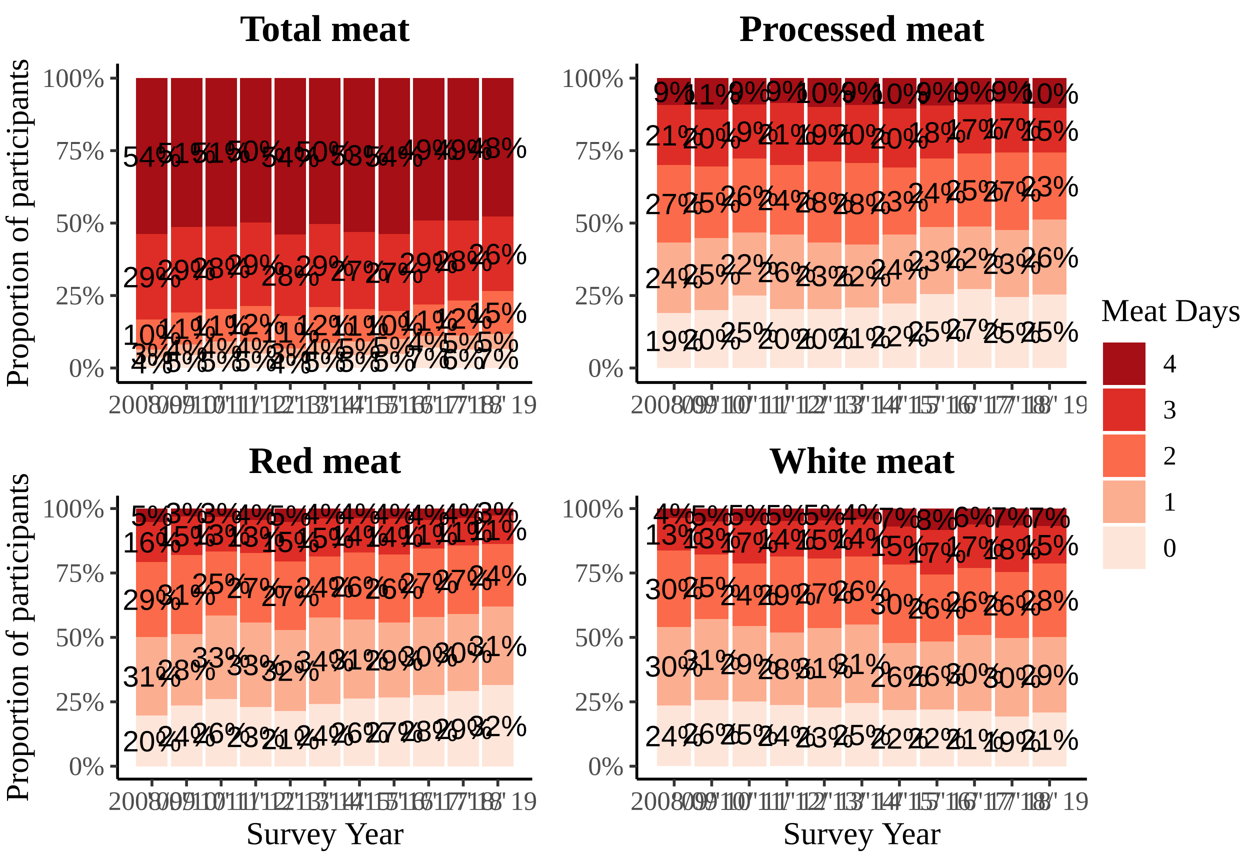



```r
#plot titles
plot1 <- plot1 + ggtitle("Total meat") + theme(plot.title = element_text(hjust = 0.5, face = "bold"))
plot2 <- plot2 + ggtitle("Processed meat") + theme(plot.title = element_text(hjust = 0.5, face = "bold"
plot3 <- plot3 + ggtitle("Red meat") + theme(plot.title = element_text(hjust = 0.5, face = "bold"), lege
plot4 <- plot4 + ggtitle("White meat") + theme(plot.title = element_text(hjust = 0.5, face = "bold"), le
#remove the x-axis label from plot1 and plot2
plot1 <- plot1 + xlab(NULL)
plot2 <- plot2 + xlab(NULL)
```

```r
#remove the y-axis label from plot2 and plot4
plot2 <- plot2 + ylab(NULL)
plot4 <- plot4 + ylab(NULL)
#take legend from plot1
plot1_legend <- cowplot::get_legend(plot1)
#remove the legend from the original plot1
plot1 <- plot1 + theme(legend.position = "none")
#combine the plots into a single 4-pane plot without the legend (I'll add it later)
combined_plot <- grid.arrange(plot1, plot2, plot3, plot4, ncol = 2, nrow = 2)
```




```r
#add the legend to the right of the combined plot
combined_plot <- grid.arrange(combined_plot, plot1_legend, ncol = 2, widths = c(8, 1))
```




```r
print(combined_plot)
```

```
## TableGrob (1 x 2) "arrange": 2 grobs
##   z     cells    name                grob
## 1 1 (1-1,1-1) arrange   gtable[arrange]
## 2 2 (1-1,2-2) arrange gtable[guide-box]
```

APPENDICES

SI TABLE 1

Analysis by standard meal time (SMT) - breakfast, lunch, dinner

```
#set survey year to factor for regression analysis
dat$SurveyYear <- as.factor(dat$SurveyYear)
dat$fpc <- 15332
dat.design <-
  svydesign(
    id = ~area,
    strata = ~astrata5,
    data = dat,
    weights = ~wti,
    fpc = ~fpc
  )

#create summary function for linear regression results
lm_summary <- function(response_var, design) {
  #define the model formula dynamically
  model_formula <- as.formula(paste(response_var, "~ SurveyYear"))

  #fit the model
  model <- svyglm(model_formula, design = design)

  #calculate the sum between the intercept and SurveyYearX coefficients
  diff <- coef(model)["(Intercept)"] + coef(model)[-1]

  #calculate the confidence intervals
  conf_int <- confint(model)

  #calculate the confidence intervals for the differences
  diff_conf_int <- conf_int["(Intercept)",] + conf_int[-1,]

  #create a new "summary" object
  summary_obj <- summary(model)

  #calculate the t-values and p-values
  t_values <- coef(model) / summary_obj$coefficients[, "Std. Error"]
  p_values <- 2 * pt(-abs(t_values), df.residual(model))

  #replace the coefficients and confidence intervals with the calculated values
  summary_obj$coefficients <- rbind(c(coef(model)["(Intercept)"], summary_obj$coefficients[1, "Std. Err
                                    cbind(diff,
                                          summary_obj$coefficients[-1, "Std. Error"],
                                          diff_conf_int,
                                          p_values[-1]))

  #update column names
  colnames(summary_obj$coefficients) <- c("Coef", "Std. Error", "2.5 %", "97.5 %", "Pr(>|t|)")

  #include the significance stars
  signif.stars <- options("show.signif.stars")
  if (is.logical(signif.stars) && signif.stars) {
    summary_obj$coefficients <- cbind(summary_obj$coefficients,
```

```
                                     summary_obj$coefficients[, "Pr(>|t|)"])
    colnames(summary_obj$coefficients)[ncol(summary_obj$coefficients)] <- " "
    summary_obj$coefficients[, " "] <- symnum(summary_obj$coefficients[, "Pr(>|t|)"],
                                              corr = FALSE, na = FALSE,
                                              cutpoints = c(0, 0.001, 0.01, 0.05, 0.1, 1),
                                              symbols = c("***", "**", "*", ".", " "))
  }

  # ADD A LITTLE 'difference of years 1 to 11 + 95%CI' ROW AT THE BOTTOM OF THE OUTPUT
  #correct the name for Intercept term [it's a wee bit messed up]
  rownames(summary_obj$coefficients)[rownames(summary_obj$coefficients) == ""] <- "Intercept"

  #calculate the difference of the intercept + SurveyYear11 coefficients
  diff_coefs <- summary_obj$coefficients["Intercept", "Coef"] - summary_obj$coefficients["SurveyYear11"

  #calculate the standard error of the difference
  se_diff <- sqrt(sum(summary_obj$coefficients[c("Intercept", "SurveyYear11"), "Std. Error"]^2))

  #calculate the confidence interval for the difference
  ci_diff <- c(diff_coefs - 1.96 * se_diff, diff_coefs + 1.96 * se_diff)

  #add these values to the summary at the bottom
  summary_obj$coefficients <- rbind(summary_obj$coefficients,
                                    c(diff_coefs, se_diff, ci_diff[1], ci_diff[2], NA))
  rownames(summary_obj$coefficients)[nrow(summary_obj$coefficients)] <- "Diff"

  #round the coefficients and the confidence intervals to 2 decimal places
  summary_obj$coefficients <- round(summary_obj$coefficients, 2)

  return(summary_obj)
}


#BREAKFAST
#overall n values (n of participants who ate breakfast)
sum(complete.cases(dat.design$variables$BsumMeatg[dat.design$variables$SurveyYear == 1])) #n values yea
```

```
## [1] 432
```

```
sum(complete.cases(dat.design$variables$BsumMeatg[dat.design$variables$SurveyYear == 11])) #n values ye
```

```
## [1] 262
```

```
##g per occasion##
lm_summary(response_var = "BsumMeatg", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
```

```
##                Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
## Intercept    102.01       6.18  89.88 114.14   <2e-16 ***
## SurveyYear2  120.77      13.57  82.02 135.27     0.17
## SurveyYear3   96.87       8.14  93.02 124.97     0.53
## SurveyYear4  104.12       8.75  74.83 109.16     0.81
## SurveyYear5   83.49       8.40  79.14 112.10     0.03 *
## SurveyYear6   99.13       9.37  68.63 105.37     0.76
## SurveyYear7   87.89       8.14  84.06 115.98     0.08 .
## SurveyYear8   87.51       8.16  59.36  91.39     0.08 .
## SurveyYear9   93.69      11.07  84.10 127.54     0.45
## SurveyYear10  91.23       9.37  60.73  97.48     0.25
## SurveyYear11  94.29       9.14  88.48 124.35     0.40
## Diff           7.72      11.04 -13.92  29.35       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 7043.318)
##
## Number of Fisher Scoring iterations: 2
```

```r
lm_summary(response_var = "BsumProcessedg", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
## Intercept     95.57       6.07  83.66 107.48   <2e-16 ***
## SurveyYear2  120.86      13.45  82.57 135.32     0.06 .
## SurveyYear3   93.16       8.10  89.19 120.95     0.77
## SurveyYear4  103.46       8.99  73.91 109.20     0.38
## SurveyYear5   87.26       8.91  81.70 116.64     0.35
## SurveyYear6   97.84       9.38  67.54 104.33     0.81
## SurveyYear7   89.03       8.04  85.17 116.70     0.42
## SurveyYear8   86.34       7.58  59.57  89.30     0.22
## SurveyYear9   88.15      11.72  77.07 123.06     0.53
## SurveyYear10  88.02       8.01  60.40  91.83     0.35
## SurveyYear11  90.09       8.58  85.16 118.83     0.52
## Diff           5.48      10.51 -15.12  26.09       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 6160.741)
##
## Number of Fisher Scoring iterations: 2
```

```r
lm_summary(response_var = "BsumRedg", design = dat.design)
```

```
##
## Call:
```

```
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
## Intercept     54.24      11.55  31.48  77.00   <2e-16 ***
## SurveyYear2   50.47      17.35  -6.46  61.88     0.83
## SurveyYear3   57.04      16.42  47.45 112.14     0.86
## SurveyYear4   28.88      12.16 -17.83  30.07     0.04 *
## SurveyYear5   29.21      12.72  26.91  77.03     0.05 *
## SurveyYear6   43.69      14.70  -8.03  49.90     0.47
## SurveyYear7   20.27      12.22  18.96  67.10     0.01 **
## SurveyYear8   28.16      12.38 -18.99  29.80     0.04 *
## SurveyYear9   53.16      19.66  37.19 114.64     0.96
## SurveyYear10  34.12      14.24 -16.68  39.42     0.16
## SurveyYear11  41.71      12.64  39.56  89.37     0.32
## Diff          12.53      17.12 -21.03  46.09       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2568.8)
##
## Number of Fisher Scoring iterations: 2
```

```r
lm_summary(response_var = "BsumWhiteg", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
## Intercept     90.05      25.19  40.46 139.65   <2e-16 ***
## SurveyYear2   43.34      27.25 -59.90  47.40     0.09 .
## SurveyYear3   67.72      28.10  62.00 172.64     0.43
## SurveyYear4   55.65      26.53 -46.17  58.29     0.20
## SurveyYear5   43.39      26.97  39.90 146.08     0.08 .
## SurveyYear6   62.82      28.44 -42.76  69.22     0.34
## SurveyYear7   42.91      26.05  41.23 143.79     0.07 .
## SurveyYear8   57.07      30.68 -52.92  67.88     0.28
## SurveyYear9   61.49      27.49  56.97 165.20     0.30
## SurveyYear10  62.69      29.72 -45.41  71.60     0.36
## SurveyYear11  64.94      28.38  58.67 170.41     0.38
## Diff          25.11      37.95 -49.27  99.49       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 4691.924)
```

```
##
## Number of Fisher Scoring iterations: 2
```

```r
lm_summary(response_var = "BokajGrams", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
## Intercept    430.43       8.19 414.37 446.49   <2e-16 ***
## SurveyYear2  421.40      12.37 381.08 429.61     0.47
## SurveyYear3  417.50      12.15 409.72 457.39     0.29
## SurveyYear4  441.34      11.20 403.31 447.25     0.33
## SurveyYear5  442.60      12.02 435.08 482.22     0.31
## SurveyYear6  454.30      12.46 413.81 462.69     0.06 .
## SurveyYear7  457.56      12.03 450.02 497.21     0.02 *
## SurveyYear8  443.67      10.46 407.09 448.13     0.21
## SurveyYear9  450.53      12.73 441.62 491.56     0.11
## SurveyYear10 472.56      12.96 431.09 481.91   <2e-16 ***
## SurveyYear11 480.11      14.98 466.79 525.54   <2e-16 ***
## Diff         -49.68      17.07 -83.13 -16.22       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 45397.07)
##
## Number of Fisher Scoring iterations: 2
```

```r
#LUNCH
#overall n values (n of participants who ate lunch)
sum(complete.cases(dat.design$variables$LsumMeatg[dat.design$variables$SurveyYear == 1])) #n values yea
```

```
## [1] 1337
```

```r
sum(complete.cases(dat.design$variables$LsumMeatg[dat.design$variables$SurveyYear == 11])) #n values ye
```

```
## [1] 835
```

```r
##g per occasion##
lm_summary(response_var = "LsumMeatg", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
```

```
## Intercept    137.62       4.35 129.09 146.16   <2e-16 ***
## SurveyYear2  131.77       7.34 108.83 137.63     0.43
## SurveyYear3  132.79       6.46 128.66 154.00     0.45
## SurveyYear4  125.21       5.89 105.14 128.22     0.04 *
## SurveyYear5  129.04       6.10 125.61 149.54     0.16
## SurveyYear6  135.15       6.74 113.40 139.82     0.71
## SurveyYear7  144.49       7.19 138.93 167.13     0.34
## SurveyYear8  131.10       6.07 110.66 134.47     0.28
## SurveyYear9  132.93       7.28 127.19 155.73     0.52
## SurveyYear10 127.88       6.54 106.51 132.18     0.14
## SurveyYear11 117.94       5.97 114.77 138.19   <2e-16 ***
## Diff          19.68       7.39   5.20  34.16       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 10068.71)
##
## Number of Fisher Scoring iterations: 2
```

```
lm_summary(response_var = "LsumProcessedg", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##               Coef Std. Error 2.5 % 97.5 % Pr(>|t|)
## Intercept    81.77       4.15 73.62  89.92   <2e-16 ***
## SurveyYear2  84.86       5.10 66.71  86.72     0.54
## SurveyYear3  85.53       5.39 83.10 104.25     0.49
## SurveyYear4  73.86       5.18 55.56  75.87     0.13
## SurveyYear5  80.10       5.32 77.81  98.69     0.75
## SurveyYear6  84.59       6.01 64.65  88.24     0.64
## SurveyYear7  79.27       6.01 75.62  99.20     0.68
## SurveyYear8  75.37       5.05 57.32  77.12     0.21
## SurveyYear9  74.51       5.58 71.72  93.60     0.19
## SurveyYear10 71.25       5.20 52.90  73.31     0.04 *
## SurveyYear11 68.90       4.91 67.42  86.67     0.01 **
## Diff         12.87       6.43  0.27  25.47       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 3848.89)
##
## Number of Fisher Scoring iterations: 2
```

```
lm_summary(response_var = "LsumRedg", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
```

```
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##               Coef Std. Error 2.5 % 97.5 % Pr(>|t|)
## Intercept    95.84       5.93 84.21 107.47   <2e-16 ***
## SurveyYear2  68.66       7.22 42.87  71.20   <2e-16 ***
## SurveyYear3  84.27       8.61 79.02 112.79     0.18
## SurveyYear4  80.96       8.21 53.23  85.43     0.07 .
## SurveyYear5  76.52       7.75 72.94 103.36     0.01 **
## SurveyYear6  76.66       8.01 49.32  80.75     0.02 *
## SurveyYear7  90.51       8.17 86.11 118.16     0.51
## SurveyYear8  76.33       7.31 50.35  79.05     0.01 **
## SurveyYear9  79.66       8.26 75.09 107.50     0.05 *
## SurveyYear10 79.10       8.03 51.72  83.22     0.04 *
## SurveyYear11 77.74       8.23 73.22 105.51     0.03 *
## Diff         18.11      10.14 -1.78  37.99       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 6266.171)
##
## Number of Fisher Scoring iterations: 2
```

```
lm_summary(response_var = "LsumWhiteg", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
## Intercept     81.45       4.29  73.03  89.87   <2e-16 ***
## SurveyYear2   91.62       8.86  65.82 100.59     0.25
## SurveyYear3   98.43       7.72  91.71 121.98     0.03 *
## SurveyYear4   84.24       6.02  64.01  87.62     0.64
## SurveyYear5   90.84       6.53  86.46 112.07     0.15
## SurveyYear6   90.19       7.73  66.61  96.92     0.26
## SurveyYear7   98.13       7.00  92.83 120.28     0.02 *
## SurveyYear8  100.58       6.34  79.72 104.60   <2e-16 ***
## SurveyYear9   99.38       8.69  90.76 124.84     0.04 *
## SurveyYear10  94.34       7.53  71.16 100.69     0.09 .
## SurveyYear11  90.21       6.64  85.60 111.66     0.19
## Diff          -8.76       7.91 -24.26   6.74       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 6643.146)
##
```

```
## Number of Fisher Scoring iterations: 2
```

```r
lm_summary(response_var = "LokajGrams", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                 Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
## Intercept    525.93      10.49 505.36 546.50   <2e-16 ***
## SurveyYear2  494.48      13.45 447.53 500.30     0.02 *
## SurveyYear3  517.26      14.33 509.72 565.94     0.55
## SurveyYear4  518.00      13.48 470.99 523.88     0.56
## SurveyYear5  509.04      13.53 503.07 556.15     0.21
## SurveyYear6  520.58      14.17 472.22 527.80     0.71
## SurveyYear7  540.44      14.38 532.82 589.21     0.31
## SurveyYear8  516.15      14.58 466.99 524.17     0.50
## SurveyYear9  539.30      15.20 530.06 589.68     0.38
## SurveyYear10 546.22      14.89 496.45 554.85     0.17
## SurveyYear11 531.22      15.14 522.09 581.49     0.73
## Diff          -5.29      18.42 -41.39  30.82       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 53417.19)
##
## Number of Fisher Scoring iterations: 2
```

```r
#DINNER
#overall n values (n of participants who ate dinner)
sum(complete.cases(dat.design$variables$DsumMeatg[dat.design$variables$SurveyYear == 1])) #n values yea
```

```
## [1] 1506
```

```r
sum(complete.cases(dat.design$variables$DsumMeatg[dat.design$variables$SurveyYear == 11])) #n values ye
```

```
## [1] 976
```

```r
##g per occasion##
lm_summary(response_var = "DsumMeatg", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                 Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
## Intercept    284.89       7.09 270.99 298.79   <2e-16 ***
```

```
## SurveyYear2  274.78      10.35 240.58 281.17     0.33
## SurveyYear3  279.72      10.31 273.40 313.84     0.62
## SurveyYear4  265.14       9.52 232.56 269.91     0.04 *
## SurveyYear5  259.47       9.69 254.36 292.38     0.01 **
## SurveyYear6  253.91       9.57 221.24 258.78   <2e-16 ***
## SurveyYear7  272.15      10.80 264.88 307.23     0.24
## SurveyYear8  256.03      10.34 221.84 262.41     0.01 **
## SurveyYear9  264.12       9.78 258.85 297.20     0.03 *
## SurveyYear10 246.78      10.12 213.04 252.72   <2e-16 ***
## SurveyYear11 241.52      10.79 234.25 276.58   <2e-16 ***
## Diff          43.37      12.91  18.07  68.68       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 28768.21)
##
## Number of Fisher Scoring iterations: 2
```

```r
lm_summary(response_var = "DsumProcessedg", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
## Intercept    105.29       5.49  94.52 116.07   <2e-16 ***
## SurveyYear2   98.64       6.97  74.20 101.53     0.34
## SurveyYear3  118.23       8.26 112.80 145.21     0.12
## SurveyYear4  100.37       6.56  76.72 102.47     0.45
## SurveyYear5   96.65       7.30  93.10 121.75     0.24
## SurveyYear6   99.95       7.49  74.48 103.86     0.48
## SurveyYear7   97.27       7.45  93.45 122.65     0.28
## SurveyYear8   90.48       7.00  65.98  93.43     0.03 *
## SurveyYear9   82.71       7.13  79.51 107.46   <2e-16 ***
## SurveyYear10  83.35       7.19  58.46  86.68   <2e-16 ***
## SurveyYear11  93.32       8.12  88.17 120.01     0.14
## Diff          11.98       9.80  -7.23  31.19       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 7307.869)
##
## Number of Fisher Scoring iterations: 2
```

```r
lm_summary(response_var = "DsumRedg", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
```

```
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                 Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
## Intercept    160.11       5.99 148.36 171.86   <2e-16 ***
## SurveyYear2  147.78       8.33 119.68 152.37     0.14
## SurveyYear3  154.29       8.92 148.55 183.53     0.51
## SurveyYear4  140.44       7.82 113.35 144.02     0.01 **
## SurveyYear5  143.42       8.33 138.83 171.51     0.05 *
## SurveyYear6  128.53       8.15 100.79 132.76   <2e-16 ***
## SurveyYear7  140.69       8.92 134.94 169.95     0.03 *
## SurveyYear8  132.03       8.38 103.85 136.71   <2e-16 ***
## SurveyYear9  137.42       8.40 132.70 165.65     0.01 **
## SurveyYear10 129.44       8.23 101.54 133.83   <2e-16 ***
## SurveyYear11 119.11       8.08 115.02 146.71   <2e-16 ***
## Diff          41.00      10.06  21.28  60.71       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 13787.54)
##
## Number of Fisher Scoring iterations: 2
```

```
lm_summary(response_var = "DsumWhiteg", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                 Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
## Intercept    146.25       5.55 135.36 157.14   <2e-16 ***
## SurveyYear2  153.96       9.98 123.49 162.65     0.44
## SurveyYear3  149.09       7.63 145.01 174.96     0.71
## SurveyYear4  149.55       7.31 124.33 152.99     0.65
## SurveyYear5  136.03       7.43 132.34 161.49     0.17
## SurveyYear6  146.74       8.31 119.55 152.16     0.95
## SurveyYear7  166.69       8.11 161.68 193.48     0.01 **
## SurveyYear8  162.19       8.65 134.33 168.28     0.07 .
## SurveyYear9  155.85       8.36 150.33 183.14     0.25
## SurveyYear10 148.12       8.27 121.01 153.45     0.82
## SurveyYear11 148.28       9.08 141.36 176.97     0.82
## Diff          -2.03      10.64 -22.89  18.83       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 13267.58)
##
## Number of Fisher Scoring iterations: 2
```

```r
lm_summary(response_var = "DokajGrams", design = dat.design)
```

```
##
## Call:
## svyglm(formula = model_formula, design = design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                Coef Std. Error  2.5 % 97.5 % Pr(>|t|)
## Intercept    564.17       9.33 545.87 582.47   <2e-16 ***
## SurveyYear2  566.14      12.97 522.41 573.28     0.88
## SurveyYear3  570.73      13.80 561.97 616.09     0.63
## SurveyYear4  573.56      12.77 530.22 580.31     0.46
## SurveyYear5  578.23      14.04 568.99 624.06     0.32
## SurveyYear6  579.30      13.73 534.08 587.93     0.27
## SurveyYear7  576.01      13.88 567.07 621.53     0.39
## SurveyYear8  555.25      13.29 510.89 563.01     0.50
## SurveyYear9  567.52      13.97 558.41 613.22     0.81
## SurveyYear10 588.84      14.73 541.66 599.43     0.09 .
## SurveyYear11 587.43      15.42 575.49 635.96     0.13
## Diff         -23.26      18.02 -58.57  12.06       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 60688.95)
##
## Number of Fisher Scoring iterations: 2
```

```r
#set survey year to numeric for p-values
dat$SurveyYear <- as.numeric(dat$SurveyYear)
dat$fpc <- 15332
dat.design <-
  svydesign(
    id = ~area,
    strata = ~astrata5,
    data = dat,
    weights = ~wti,
    fpc = ~fpc
  )

#BREAKFAST
#p values
summary(svyglm(BsumMeatg ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = BsumMeatg ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 107.2247     5.0314  21.311   <2e-16 ***
## SurveyYear   -1.8350     0.7259  -2.528   0.0116 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 7109.342)
##
## Number of Fisher Scoring iterations: 2
```

```
summary(svyglm(BsumProcessedg ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = BsumProcessedg ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 105.0220     4.9797  21.090   <2e-16 ***
## SurveyYear   -1.8052     0.7009  -2.576   0.0101 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 6222.735)
##
## Number of Fisher Scoring iterations: 2
```

```
summary(svyglm(BsumRedg ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = BsumRedg ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   49.931      7.675   6.505  4.3e-10 ***
## SurveyYear    -1.752      1.058  -1.655   0.0991 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2722.475)
##
## Number of Fisher Scoring iterations: 2
```

```
summary(svyglm(BsumWhiteg ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = BsumWhiteg ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  62.3235    12.2597   5.084 6.76e-07 ***
## SurveyYear   -0.5486     1.7146  -0.320    0.749
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 4916.17)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(BokajGrams ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = BokajGrams ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 415.7866     5.8052   71.62  < 2e-16 ***
## SurveyYear    5.1144     0.9198    5.56 3.11e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 45471.18)
##
## Number of Fisher Scoring iterations: 2
```

```r
#LUNCH
#p values
summary(svyglm(LsumMeatg ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = LsumMeatg ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 135.5969     3.1322   43.29   <2e-16 ***
```

```
## SurveyYear   -0.7037     0.4630   -1.52    0.129
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 10102.16)
##
## Number of Fisher Scoring iterations: 2
```

```
summary(svyglm(LsumProcessedg ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = LsumProcessedg ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  85.7101     2.3525  36.434  < 2e-16 ***
## SurveyYear   -1.2774     0.3374  -3.786 0.000158 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 3860.249)
##
## Number of Fisher Scoring iterations: 2
```

```
summary(svyglm(LsumRedg ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = LsumRedg ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   84.721      3.704  22.874   <2e-16 ***
## SurveyYear    -0.651      0.546  -1.192    0.233
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 6328.096)
##
## Number of Fisher Scoring iterations: 2
```

```
summary(svyglm(LsumWhiteg ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = LsumWhiteg ~ SurveyYear, design = dat.design)
##
```

```
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  86.5342     3.7020  23.375   <2e-16 ***
## SurveyYear    1.0130     0.5516   1.836   0.0665 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 6670.713)
##
## Number of Fisher Scoring iterations: 2
```

```
summary(svyglm(LokajGrams ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = LokajGrams ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 506.2698     6.3455  79.784  < 2e-16 ***
## SurveyYear    2.8905     0.9818   2.944  0.00328 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 53544.88)
##
## Number of Fisher Scoring iterations: 2
```

```
#DINNER
#p values
summary(svyglm(DsumMeatg ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = DsumMeatg ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  284.556      4.710  60.419  < 2e-16 ***
## SurveyYear    -3.489      0.710  -4.914 9.74e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for gaussian family taken to be 28813.39)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(DsumProcessedg ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = DsumProcessedg ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 109.9192     3.3552  32.761  < 2e-16 ***
## SurveyYear   -2.1676     0.5019  -4.319 1.66e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 7345.867)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(DsumRedg ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = DsumRedg ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 158.6176     3.8695  40.992  < 2e-16 ***
## SurveyYear   -3.1973     0.5681  -5.628 2.12e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 13816.91)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(DsumWhiteg ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = DsumWhiteg ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 147.6464     4.0908  36.093   <2e-16 ***
## SurveyYear    0.6198     0.6184   1.002    0.316
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 13323.14)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(DokajGrams ~ SurveyYear, dat.design))
```

```
##
## Call:
## svyglm(formula = DokajGrams ~ SurveyYear, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 564.8057     6.2631  90.180   <2e-16 ***
## SurveyYear    1.4086     0.9874   1.427    0.154
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 60753.64)
##
## Number of Fisher Scoring iterations: 2
```

SI TABLE 2a

change in meat consumption behaviours by sex

```
#set survey year as factor for regression analysis
dat$SurveyYear <- as.factor(dat$SurveyYear)
dat$fpc <- 15332
dat.design <-
  svydesign(
    id = ~area,
    strata = ~astrata5,
    data = dat,
    weights = ~wti,
    fpc = ~fpc
  )

#create summary functions for exponentiation and glm regression analyses
exp_interaction_CI_sex <- function(response_var, design) {
  model_formula <- as.formula(paste(response_var, "~ SurveyYear + Sex + SurveyYear * Sex"))
  model <- svyglm(model_formula, family = poisson(link = "log"), design = design)
  model_summary <- summary(model)
  betas <- coef(model)
  se <- coef(model_summary)[, "Std. Error"]
  #combinations of coefficients
  b_combinations <- c(
    betas["(Intercept)"],
    betas["(Intercept)"] + betas["SurveyYear11"],
    betas["(Intercept)"] + betas["SexF"],
    betas["(Intercept)"] + betas["SexF"] + betas["SurveyYear11"] + betas["SurveyYear11:SexF"]
  )
  #standard errors for combinations
  se_combinations <- c(
    se["(Intercept)"],
    sqrt(se["(Intercept)"]^2 + se["SurveyYear11"]^2),
    sqrt(se["(Intercept)"]^2 + se["SexF"]^2),
    sqrt(se["(Intercept)"]^2 + se["SexF"]^2 + se["SurveyYear11"]^2 + se["SurveyYear11:SexF"]^2)
  )
  #exponentiated coefficients and CI for combinations
  exp_coef <- round(exp(b_combinations), 2)
  lower_bound <- round(exp(b_combinations - 1.96 * se_combinations), 2)
  upper_bound <- round(exp(b_combinations + 1.96 * se_combinations), 2)

  #difference between groups
  diff_coefs <- c(exp_coef[1] - exp_coef[2], exp_coef[3] - exp_coef[4])

  #standard error of the difference
  se_diff <- c(sqrt(sum(se_combinations[1:2]^2)), sqrt(sum(se_combinations[3:4]^2)))

  #confidence interval for the difference
  ci_diff_lower <- c(diff_coefs[1] - 1.96 * se_diff[1], diff_coefs[2] - 1.96 * se_diff[2])
  ci_diff_upper <- c(diff_coefs[1] + 1.96 * se_diff[1], diff_coefs[2] + 1.96 * se_diff[2])

  #data frame to present the results
  result_table <- rbind(
```

```r
    data.frame(
      Group = c("M_Y1", "M_Y11", "F_Y1", "F_Y11"),
      Beta = exp_coef,
      Lower = lower_bound,
      Upper = upper_bound,
      stringsAsFactors = FALSE
    ),
    data.frame(
      Group = c("Diff_M", "Diff_F"),
      Beta = round(diff_coefs, 2),
      Lower = round(ci_diff_lower, 2),
      Upper = round(ci_diff_upper, 2),
      stringsAsFactors = FALSE
    )
  )
  return(result_table)
}
glm_interaction_CI_sex <- function(response_var, design) {
  model_formula <- as.formula(paste(response_var, "~ SurveyYear + Sex + SurveyYear * Sex"))
  model <- svyglm(model_formula, design = design)
  model_summary <- summary(model)
  betas <- coef(model)
  se <- coef(model_summary)[, "Std. Error"]
  #combinations of coefficients
  b_combinations <- c(
    betas["(Intercept)"],
    betas["(Intercept)"] + betas["SurveyYear11"],
    betas["(Intercept)"] + betas["SexF"],
    betas["(Intercept)"] + betas["SexF"] + betas["SurveyYear11"] + betas["SurveyYear11:SexF"]
  )
  #standard errors for combinations
  se_combinations <- c(
    se["(Intercept)"],
    sqrt(se["(Intercept)"]^2 + se["SurveyYear11"]^2),
    sqrt(se["(Intercept)"]^2 + se["SexF"]^2),
    sqrt(se["(Intercept)"]^2 + se["SexF"]^2 + se["SurveyYear11"]^2 + se["SurveyYear11:SexF"]^2)
  )
  #CI for combinations
  lower_bound <- round(b_combinations - 1.96 * se_combinations, 2)
  upper_bound <- round(b_combinations + 1.96 * se_combinations, 2)

  #difference between groups
  diff_coefs <- c(b_combinations[1] - b_combinations[2], b_combinations[3] - b_combinations[4])

  #standard error of the difference
  se_diff <- c(sqrt(sum(se_combinations[1:2]^2)), sqrt(sum(se_combinations[3:4]^2)))

  #confidence interval for the difference
  ci_diff_lower <- c(diff_coefs[1] - 1.96 * se_diff[1], diff_coefs[2] - 1.96 * se_diff[2])
  ci_diff_upper <- c(diff_coefs[1] + 1.96 * se_diff[1], diff_coefs[2] + 1.96 * se_diff[2])

  #data frame to present the results
  result_table <- rbind(
```

```r
    data.frame(
      Group = c("M_Y1", "M_Y11", "F_Y1", "F_Y11"),
      Beta = b_combinations,
      Lower = lower_bound,
      Upper = upper_bound,
      stringsAsFactors = FALSE
    ),
    data.frame(
      Group = c("Diff_M", "Diff_F"),
      Beta = round(diff_coefs, 2),
      Lower = round(ci_diff_lower, 2),
      Upper = round(ci_diff_upper, 2),
      stringsAsFactors = FALSE
    )
  )
  return(result_table)
}

#days
exp_interaction_CI_sex(response_var = "MeatDays", design = dat.design)
```

```
##    Group Beta Lower Upper
## 1   M_Y1 3.34  3.24  3.45
## 2  M_Y11 3.13  2.93  3.35
## 3   F_Y1 3.20  3.03  3.39
## 4  F_Y11 2.93  2.62  3.28
## 5 Diff_M 0.21  0.14  0.28
## 6 Diff_F 0.27  0.14  0.40
```

```r
exp_interaction_CI_sex(response_var = "ProcessedDays", design = dat.design)
```

```
##    Group Beta Lower Upper
## 1   M_Y1 1.89  1.75  2.05
## 2  M_Y11 1.75  1.51  2.03
## 3   F_Y1 1.66  1.45  1.88
## 4  F_Y11 1.45  1.14  1.85
## 5 Diff_M 0.14 -0.03  0.31
## 6 Diff_F 0.21 -0.06  0.48
```

```r
exp_interaction_CI_sex(response_var = "RedDays", design = dat.design)
```

```
##    Group Beta Lower Upper
## 1   M_Y1 1.61  1.49  1.74
## 2  M_Y11 1.27  1.10  1.48
## 3   F_Y1 1.51  1.32  1.74
## 4  F_Y11 1.18  0.92  1.53
## 5 Diff_M 0.34  0.17  0.51
## 6 Diff_F 0.33  0.04  0.62
```

```r
exp_interaction_CI_sex(response_var = "WhiteDays", design = dat.design)
```

```
##    Group  Beta Lower Upper
## 1   M_Y1  1.40  1.29  1.51
## 2  M_Y11  1.65  1.43  1.91
## 3   F_Y1  1.44  1.26  1.66
## 4  F_Y11  1.49  1.18  1.90
```

```
## 5 Diff_M -0.25 -0.42 -0.08
## 6 Diff_F -0.05 -0.33  0.23
```

```r
exp_interaction_CI_sex(response_var = "NoMeatDays", design = dat.design)
```

```
##    Group  Beta Lower Upper
## 1   M_Y1  0.66  0.56  0.77
## 2  M_Y11  0.87  0.65  1.15
## 3   F_Y1  0.80  0.61  1.04
## 4  F_Y11  1.07  0.67  1.70
## 5 Diff_M -0.21 -0.54  0.12
## 6 Diff_F -0.27 -0.80  0.26
```

```r
#occasions
exp_interaction_CI_sex(response_var = "avgMeatokaj", design = dat.design)
```

```
##    Group Beta Lower Upper
## 1   M_Y1 1.31  1.25  1.38
## 2  M_Y11 1.25  1.13  1.37
## 3   F_Y1 1.17  1.08  1.27
## 4  F_Y11 1.03  0.88  1.20
## 5 Diff_M 0.06 -0.05  0.17
## 6 Diff_F 0.14 -0.04  0.32
```

```r
exp_interaction_CI_sex(response_var = "avgProcessedokaj", design = dat.design)
```

```
##    Group Beta Lower Upper
## 1   M_Y1 0.59  0.54  0.64
## 2  M_Y11 0.57  0.48  0.68
## 3   F_Y1 0.50  0.43  0.58
## 4  F_Y11 0.43  0.32  0.57
## 5 Diff_M 0.02 -0.17  0.21
## 6 Diff_F 0.07 -0.25  0.39
```

```r
exp_interaction_CI_sex(response_var = "avgRedokaj", design = dat.design)
```

```
##    Group Beta Lower Upper
## 1   M_Y1 0.47  0.42  0.52
## 2  M_Y11 0.35  0.29  0.43
## 3   F_Y1 0.41  0.35  0.50
## 4  F_Y11 0.32  0.23  0.43
## 5 Diff_M 0.12 -0.10  0.34
## 6 Diff_F 0.09 -0.27  0.45
```

```r
exp_interaction_CI_sex(response_var = "avgWhiteokaj", design = dat.design)
```

```
##    Group  Beta Lower Upper
## 1   M_Y1  0.41  0.37  0.46
## 2  M_Y11  0.49  0.41  0.58
## 3   F_Y1  0.40  0.34  0.47
## 4  F_Y11  0.42  0.31  0.56
## 5 Diff_M -0.08 -0.29  0.13
## 6 Diff_F -0.02 -0.36  0.32
```

```r
#portion size
glm_interaction_CI_sex(response_var = "gperokajMeat", design = dat.design)
```

```
##    Group     Beta  Lower   Upper
```

```
## 1   M_Y1 96.19095  90.25 102.14
## 2  M_Y11 81.27507  71.75  90.80
## 3   F_Y1 75.87541  67.22  84.53
## 4  F_Y11 71.40154  56.60  86.20
## 5 Diff_M 14.92000   3.69  26.14
## 6 Diff_F  4.47000 -12.67  21.62
```

```r
glm_interaction_CI_sex(response_var = "gperokajProcessed", design = dat.design)
```

```
##    Group     Beta  Lower Upper
## 1   M_Y1 74.34096  66.83 81.86
## 2  M_Y11 58.34123  46.54 70.15
## 3   F_Y1 52.57188  41.66 63.49
## 4  F_Y11 46.98301  29.40 64.56
## 5 Diff_M 16.00000   2.00 29.99
## 6 Diff_F  5.59000 -15.11 26.28
```

```r
glm_interaction_CI_sex(response_var = "gperokajRed", design = dat.design)
```

```
##    Group      Beta  Lower  Upper
## 1   M_Y1 101.58531  89.74 113.43
## 2  M_Y11  73.37146  54.82  91.92
## 3   F_Y1  78.56700  60.94  96.19
## 4  F_Y11  66.24238  38.10  94.38
## 5 Diff_M  28.21000   6.20  50.22
## 6 Diff_F  12.32000 -20.88  45.53
```

```r
glm_interaction_CI_sex(response_var = "gperokajWhite", design = dat.design)
```

```
##    Group     Beta  Lower  Upper
## 1   M_Y1 92.87735  84.64 101.11
## 2  M_Y11 86.21926  72.68  99.76
## 3   F_Y1 77.04011  64.69  89.39
## 4  F_Y11 73.75208  52.80  94.70
## 5 Diff_M  6.66000  -9.19  22.50
## 6 Diff_F  3.29000 -21.03  27.60
```

```r
#set survey year to numeric for p-values
dat$SurveyYear <- as.numeric(dat$SurveyYear)
dat$fpc <- 15332
dat.design <-
  svydesign(
    id = ~area,
    strata = ~astrata5,
    data = dat,
    weights = ~wti,
    fpc = ~fpc
  )

#p values
summary(svyglm(MeatDays ~ SurveyYear + Sex + SurveyYear * Sex, family = poisson(link = "log"), dat.desi
```

```
##
## Call:
## svyglm(formula = MeatDays ~ SurveyYear + Sex + SurveyYear * Sex,
##     design = dat.design, family = poisson(link = "log"))
##
```

```
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.213482   0.011200 108.344  < 2e-16 ***
## SurveyYear      -0.003236   0.001854  -1.746    0.081 .
## SexF            -0.065811   0.016364  -4.022 6.02e-05 ***
## SurveyYear:SexF -0.003752   0.002590  -1.449    0.148
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3793134)
##
## Number of Fisher Scoring iterations: 4
```

```
summary(svyglm(ProcessedDays ~ SurveyYear + Sex + SurveyYear * Sex, family = poisson(link = "log"), dat
```

```
##
## Call:
## svyglm(formula = ProcessedDays ~ SurveyYear + Sex + SurveyYear *
##     Sex, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.656796   0.026454  24.828  < 2e-16 ***
## SurveyYear      -0.005655   0.004226  -1.338    0.181
## SexF            -0.160010   0.033976  -4.709 2.68e-06 ***
## SurveyYear:SexF -0.008349   0.005362  -1.557    0.120
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.9385183)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(RedDays ~ SurveyYear + Sex + SurveyYear * Sex, family = poisson(link = "log"), dat.desig
```

```
##
## Call:
## svyglm(formula = RedDays ~ SurveyYear + Sex + SurveyYear * Sex,
##     design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.477897   0.027626  17.299  < 2e-16 ***
```

```
## SurveyYear      -0.012592   0.004238  -2.972  0.00300 **
## SexF            -0.100922   0.038600  -2.615  0.00901 **
## SurveyYear:SexF -0.007716   0.005950  -1.297  0.19482
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.8818821)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(WhiteDays ~ SurveyYear + Sex + SurveyYear * Sex, family = poisson(link = "log"), dat.des
```

```
##
## Call:
## svyglm(formula = WhiteDays ~ SurveyYear + Sex + SurveyYear *
##     Sex, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.343586   0.028480  12.064  < 2e-16 ***
## SurveyYear       0.016909   0.004201   4.025 5.94e-05 ***
## SexF            -0.033175   0.038349  -0.865    0.387
## SurveyYear:SexF -0.004835   0.005569  -0.868    0.385
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.8848693)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(NoMeatDays ~ SurveyYear + Sex + SurveyYear * Sex, family = poisson(link = "log"), dat.de
```

```
##
## Call:
## svyglm(formula = NoMeatDays ~ SurveyYear + Sex + SurveyYear *
##     Sex, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -0.450736   0.055534  -8.116 8.96e-16 ***
## SurveyYear       0.015329   0.008709   1.760   0.0786 .
## SexF             0.295537   0.068056   4.343 1.49e-05 ***
## SurveyYear:SexF  0.006394   0.010252   0.624   0.5329
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1.433568)
```

```
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(avgMeatokaj ~ SurveyYear + Sex + SurveyYear * Sex, family = poisson(link = "log"), dat.de
```

```
##
## Call:
## svyglm(formula = avgMeatokaj ~ SurveyYear + Sex + SurveyYear *
##     Sex, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.275355   0.017836  15.438  < 2e-16 ***
## SurveyYear      -0.002765   0.002792  -0.990    0.322
## SexF            -0.148403   0.023202  -6.396 2.04e-10 ***
## SurveyYear:SexF -0.004871   0.003613  -1.348    0.178
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.2754542)
##
## Number of Fisher Scoring iterations: 4
```

```
summary(svyglm(avgProcessedokaj ~ SurveyYear + Sex + SurveyYear * Sex, family = poisson(link = "log"), d
```

```
##
## Call:
## svyglm(formula = avgProcessedokaj ~ SurveyYear + Sex + SurveyYear *
##     Sex, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -0.492909   0.031764 -15.518  < 2e-16 ***
## SurveyYear      -0.005776   0.004946  -1.168    0.243
## SexF            -0.231037   0.039421  -5.861 5.49e-09 ***
## SurveyYear:SexF -0.008193   0.006207  -1.320    0.187
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3690807)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(avgRedokaj ~ SurveyYear + Sex + SurveyYear * Sex, family = poisson(link = "log"), dat.des
```

```
##
## Call:
## svyglm(formula = avgRedokaj ~ SurveyYear + Sex + SurveyYear *
```

```
##     Sex, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -0.772880   0.033143 -23.320  < 2e-16 ***
## SurveyYear      -0.014713   0.004946  -2.974 0.002976 **
## SexF            -0.150589   0.044525  -3.382 0.000735 ***
## SurveyYear:SexF -0.004975   0.006654  -0.748 0.454747
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.2948715)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(avgWhiteokaj ~ SurveyYear + Sex + SurveyYear * Sex, family = poisson(link = "log"), dat.d
```

```
##
## Call:
## svyglm(formula = avgWhiteokaj ~ SurveyYear + Sex + SurveyYear *
##     Sex, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -0.896935   0.035765 -25.078  < 2e-16 ***
## SurveyYear       0.018112   0.005208   3.478 0.000517 ***
## SexF            -0.072561   0.045489  -1.595 0.110863
## SurveyYear:SexF -0.003796   0.006633  -0.572 0.567186
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3285289)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(gperokajMeat ~ SurveyYear + Sex + SurveyYear * Sex, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajMeat ~ SurveyYear + Sex + SurveyYear *
##     Sex, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
```

```
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       93.9013     1.6653  56.386  < 2e-16 ***
## SurveyYear        -1.0562     0.2416  -4.371 1.31e-05 ***
## SexF             -17.3667     1.9746  -8.795  < 2e-16 ***
## SurveyYear:SexF    0.6604     0.2967   2.226   0.0261 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1313.336)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(gperokajProcessed ~ SurveyYear + Sex + SurveyYear * Sex, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajProcessed ~ SurveyYear + Sex + SurveyYear *
##     Sex, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       73.7979     2.0275  36.399  < 2e-16 ***
## SurveyYear        -1.6498     0.2868  -5.752 1.04e-08 ***
## SexF             -17.6307     2.3694  -7.441 1.56e-13 ***
## SurveyYear:SexF    0.8108     0.3394   2.389    0.017 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1415.647)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(gperokajRed ~ SurveyYear + Sex + SurveyYear * Sex, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajRed ~ SurveyYear + Sex + SurveyYear *
##     Sex, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       95.7299     3.1769  30.133  < 2e-16 ***
## SurveyYear        -1.9474     0.4561  -4.270 2.06e-05 ***
## SexF             -19.0487     3.6663  -5.196 2.28e-07 ***
## SurveyYear:SexF    1.1011     0.5410   2.035    0.042 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for gaussian family taken to be 2957.98)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(gperokajWhite ~ SurveyYear + Sex + SurveyYear * Sex, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajWhite ~ SurveyYear + Sex + SurveyYear *
##     Sex, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      90.7172     2.5388  35.732  < 2e-16 ***
## SurveyYear       -0.2506     0.3662  -0.684 0.493918
## SexF            -10.9666     3.0766  -3.565 0.000374 ***
## SurveyYear:SexF  -0.1159     0.4459  -0.260 0.794888
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2512.907)
##
## Number of Fisher Scoring iterations: 2
```

SI TABLE 2b

change in meat consumption behaviours by age group

```r
#set survey year as factor for regression analysis
dat$SurveyYear <- as.factor(dat$SurveyYear)
dat$fpc <- 15332
dat.design <-
  svydesign(
    id = ~area,
    strata = ~astrata5,
    data = dat,
    weights = ~wti,
    fpc = ~fpc
  )

#create summary functions for exponentiation and glm regression analyses
exp_interaction_CI_age <- function(response_var, design) {
  model_formula <- as.formula(paste(response_var, "~ SurveyYear + AgeG + SurveyYear * AgeG"))
  model <- svyglm(model_formula, family = poisson(link = "log"), design = design)
  model_summary <- summary(model)
  betas <- coef(model)
  se <- coef(model_summary)[, "Std. Error"]

  # combinations of coefficients
  b_combinations <- c(
    betas["(Intercept)"],
    betas["(Intercept)"] + betas["SurveyYear11"],
    betas["(Intercept)"] + betas["AgeG1"],
    betas["(Intercept)"] + betas["AgeG1"] + betas["SurveyYear11"] + betas["SurveyYear11:AgeG1"],
    betas["(Intercept)"] + betas["AgeG2"],
    betas["(Intercept)"] + betas["AgeG2"] + betas["SurveyYear11"] + betas["SurveyYear11:AgeG2"],
    betas["(Intercept)"] + betas["AgeG4"],
    betas["(Intercept)"] + betas["AgeG4"] + betas["SurveyYear11"] + betas["SurveyYear11:AgeG4"],
    betas["(Intercept)"] + betas["AgeG5"],
    betas["(Intercept)"] + betas["AgeG5"] + betas["SurveyYear11"] + betas["SurveyYear11:AgeG5"]
  )

  # exponentiated coefficients
  exp_coef <- round(exp(b_combinations), 2)

  # standard errors for combinations
  se_combinations <- c(
    se["(Intercept)"],
    sqrt(se["(Intercept)"]^2 + se["SurveyYear11"]^2),
    se["AgeG1"],
    sqrt(se["AgeG1"]^2 + se["SurveyYear11"]^2 + se["SurveyYear11:AgeG1"]^2),
    se["AgeG2"],
    sqrt(se["AgeG2"]^2 + se["SurveyYear11"]^2 + se["SurveyYear11:AgeG2"]^2),
    se["AgeG4"],
    sqrt(se["AgeG4"]^2 + se["SurveyYear11"]^2 + se["SurveyYear11:AgeG4"]^2),
    se["AgeG5"],
    sqrt(se["AgeG5"]^2 + se["SurveyYear11"]^2 + se["SurveyYear11:AgeG5"]^2)
  )
```

```r
  # CI for combinations
  lower_bound <- round(exp_coef - 1.96 * se_combinations, 2)
  upper_bound <- round(exp_coef + 1.96 * se_combinations, 2)

  # differences of exponentiated coefficients for each pair of groups (Y1 - Y11)
  diff_coefs <- exp_coef[seq(1, length(exp_coef), 2)] - exp_coef[seq(2, length(exp_coef), 2)]

  # standard errors for the differences
  se_diff <- sqrt(se_combinations[seq(1, length(se_combinations), 2)]^2 + se_combinations[seq(2, length

  # CI for the differences
  diff_lower_bound <- round(diff_coefs - 1.96 * se_diff, 2)
  diff_upper_bound <- round(diff_coefs + 1.96 * se_diff, 2)

  # data frame to present the results
  result_table <- rbind(
    data.frame(
      Group = c("18-40_Y1", "18-40_Y11", "<10_Y1", "<10_Y11", "11-17_Y1", "11-17_Y11", "41-59_Y1", "41-5
      Beta = exp_coef,
      Lower = lower_bound,
      Upper = upper_bound
    ),
    data.frame(
      Group = c("Diff_18-40", "Diff_<10", "Diff_11-17", "Diff_41-59", "Diff_>=60"),
      Beta = round(diff_coefs, 2),
      Lower = round(diff_lower_bound, 2),
      Upper = round(diff_upper_bound, 2)
    )
  )
  return(result_table)
}
glm_interaction_CI_age <- function(response_var, design) {
  model_formula <- as.formula(paste(response_var, "~ SurveyYear + AgeG + SurveyYear * AgeG"))
  model <- svyglm(model_formula, design = design)
  model_summary <- summary(model)
  betas <- coef(model)
  se <- coef(model_summary)[, "Std. Error"]
  #combinations of coefficients
  b_combinations <- c(
    betas["(Intercept)"],
    betas["(Intercept)"] + betas["SurveyYear11"],
    betas["(Intercept)"] + betas["AgeG1"],
    betas["(Intercept)"] + betas["AgeG1"] + betas["SurveyYear11"] + betas["SurveyYear11:AgeG1"],
    betas["(Intercept)"] + betas["AgeG2"],
    betas["(Intercept)"] + betas["AgeG2"] + betas["SurveyYear11"] + betas["SurveyYear11:AgeG2"],
    betas["(Intercept)"] + betas["AgeG4"],
    betas["(Intercept)"] + betas["AgeG4"] + betas["SurveyYear11"] + betas["SurveyYear11:AgeG4"],
    betas["(Intercept)"] + betas["AgeG5"],
    betas["(Intercept)"] + betas["AgeG5"] + betas["SurveyYear11"] + betas["SurveyYear11:AgeG5"]
  )
  #standard errors for combinations
  se_combinations <- c(
    se["(Intercept)"],
```

```
    sqrt(se["(Intercept)"]^2 + se["SurveyYear11"]^2),
    se["AgeG1"],
    sqrt(se["AgeG1"]^2 + se["SurveyYear11"]^2 + se["SurveyYear11:AgeG1"]^2),
    se["AgeG2"],
    sqrt(se["AgeG2"]^2 + se["SurveyYear11"]^2 + se["SurveyYear11:AgeG2"]^2),
    se["AgeG4"],
    sqrt(se["AgeG4"]^2 + se["SurveyYear11"]^2 + se["SurveyYear11:AgeG4"]^2),
    se["AgeG5"],
    sqrt(se["AgeG5"]^2 + se["SurveyYear11"]^2 + se["SurveyYear11:AgeG5"]^2)
  )
  # CI for combinations
  lower_bound <- round(b_combinations - 1.96 * se_combinations, 2)
  upper_bound <- round(b_combinations + 1.96 * se_combinations, 2)

  #difference between groups
  diff_coefs <- b_combinations[seq(1, length(b_combinations), 2)] - b_combinations[seq(2, length(b_combi

  #standard error of the difference
  se_diff <- sqrt(se_combinations[seq(1, length(se_combinations), 2)]^2 + se_combinations[seq(2, length

  #confidence interval for the difference
  ci_diff_lower <- diff_coefs - 1.96 * se_diff
  ci_diff_upper <- diff_coefs + 1.96 * se_diff

  #data frame to present the results
  result_table <- rbind(
    data.frame(
      Group = c("18-40_Y1", "18-40_Y11", "<10_Y1", "<10_Y11", "11-17_Y1", "11-17_Y11", "41-59_Y1", "41-5
      Beta = b_combinations,
      Lower = lower_bound,
      Upper = upper_bound,
      stringsAsFactors = FALSE
    ),
    data.frame(
      Group = c("Diff_18-40", "Diff_<10", "Diff_11-17", "Diff_41-59", "Diff_>=60"),
      Beta = round(diff_coefs, 2),
      Lower = round(ci_diff_lower, 2),
      Upper = round(ci_diff_upper, 2),
      stringsAsFactors = FALSE
    )
  )
  return(result_table)
}

#days
exp_interaction_CI_age(response_var = "MeatDays", design = dat.design)
```

```
##         Group Beta Lower Upper
## 1    18-40_Y1 3.21  3.16  3.26
## 2   18-40_Y11 2.90  2.79  3.01
## 3      <10_Y1 3.29  3.23  3.35
## 4     <10_Y11 3.18  3.02  3.34
## 5    11-17_Y1 3.38  3.32  3.44
## 6   11-17_Y11 3.37  3.21  3.53
```

```
## 7    41-59_Y1 3.30  3.23  3.37
## 8   41-59_Y11 2.95  2.77  3.13
## 9      >=60_Y1 3.28  3.22  3.34
## 10   >=60_Y11 3.07  2.91  3.23
## 11 Diff_18-40 0.31  0.19  0.43
## 12   Diff_<10 0.11 -0.06  0.28
## 13 Diff_11-17 0.01 -0.16  0.18
## 14 Diff_41-59 0.35  0.16  0.54
## 15  Diff_>=60 0.21  0.04  0.38
```

```
exp_interaction_CI_age(response_var = "ProcessedDays", design = dat.design)
```

```
##          Group Beta Lower Upper
## 1    18-40_Y1 1.81  1.72  1.90
## 2   18-40_Y11 1.61  1.41  1.81
## 3      <10_Y1 1.88  1.76  2.00
## 4     <10_Y11 1.72  1.41  2.03
## 5    11-17_Y1 1.96  1.83  2.09
## 6   11-17_Y11 1.88  1.56  2.20
## 7    41-59_Y1 1.73  1.58  1.88
## 8   41-59_Y11 1.51  1.16  1.86
## 9     >=60_Y1 1.62  1.46  1.78
## 10   >=60_Y11 1.49  1.15  1.83
## 11 Diff_18-40 0.20 -0.02  0.42
## 12   Diff_<10 0.16 -0.17  0.49
## 13 Diff_11-17 0.08 -0.27  0.43
## 14 Diff_41-59 0.22 -0.16  0.60
## 15  Diff_>=60 0.13 -0.24  0.50
```

```
exp_interaction_CI_age(response_var = "RedDays", design = dat.design)
```

```
##          Group  Beta Lower Upper
## 1    18-40_Y1  1.41  1.30  1.52
## 2   18-40_Y11  1.06  0.84  1.28
## 3      <10_Y1  1.19  1.06  1.32
## 4     <10_Y11  1.07  0.75  1.39
## 5    11-17_Y1  1.33  1.17  1.49
## 6   11-17_Y11  1.41  1.05  1.77
## 7    41-59_Y1  1.74  1.59  1.89
## 8   41-59_Y11  1.17  0.82  1.52
## 9     >=60_Y1  1.86  1.71  2.01
## 10   >=60_Y11  1.49  1.15  1.83
## 11 Diff_18-40  0.35  0.10  0.60
## 12   Diff_<10  0.12 -0.23  0.47
## 13 Diff_11-17 -0.08 -0.47  0.31
## 14 Diff_41-59  0.57  0.20  0.94
## 15  Diff_>=60  0.37  0.00  0.74
```

```
exp_interaction_CI_age(response_var = "WhiteDays", design = dat.design)
```

```
##          Group  Beta Lower Upper
## 1    18-40_Y1  1.64  1.54  1.74
## 2   18-40_Y11  1.78  1.60  1.96
## 3      <10_Y1  1.45  1.32  1.58
## 4     <10_Y11  1.66  1.39  1.93
## 5    11-17_Y1  1.68  1.55  1.81
```

```
## 6   11-17_Y11  1.85  1.57  2.13
## 7    41-59_Y1  1.29  1.13  1.45
## 8   41-59_Y11  1.47  1.15  1.79
## 9      >=60_Y1  1.14  0.96  1.32
## 10    >=60_Y11  1.27  0.93  1.61
## 11 Diff_18-40 -0.14 -0.34  0.06
## 12   Diff_<10 -0.21 -0.51  0.09
## 13 Diff_11-17 -0.17 -0.48  0.14
## 14 Diff_41-59 -0.18 -0.54  0.18
## 15  Diff_>=60 -0.13 -0.51  0.25
```

```r
exp_interaction_CI_age(response_var = "NoMeatDays", design = dat.design)
```

```
##         Group  Beta Lower Upper
## 1    18-40_Y1  0.79  0.59  0.99
## 2   18-40_Y11  1.10  0.74  1.46
## 3      <10_Y1  0.71  0.45  0.97
## 4     <10_Y11  0.82  0.28  1.36
## 5    11-17_Y1  0.62  0.32  0.92
## 6   11-17_Y11  0.63  0.02  1.24
## 7    41-59_Y1  0.70  0.40  1.00
## 8   41-59_Y11  1.05  0.45  1.65
## 9     >=60_Y1  0.72  0.47  0.97
## 10   >=60_Y11  0.93  0.40  1.46
## 11 Diff_18-40 -0.31 -0.72  0.10
## 12   Diff_<10 -0.11 -0.71  0.49
## 13 Diff_11-17 -0.01 -0.68  0.66
## 14 Diff_41-59 -0.35 -1.02  0.32
## 15  Diff_>=60 -0.21 -0.80  0.38
```

```r
#occasions
exp_interaction_CI_age(response_var = "avgMeatokaj", design = dat.design)
```

```
##         Group  Beta Lower Upper
## 1    18-40_Y1  1.26  1.19  1.33
## 2   18-40_Y11  1.14  1.00  1.28
## 3      <10_Y1  1.23  1.15  1.31
## 4     <10_Y11  1.18  0.97  1.39
## 5    11-17_Y1  1.34  1.26  1.42
## 6   11-17_Y11  1.35  1.14  1.56
## 7    41-59_Y1  1.24  1.14  1.34
## 8   41-59_Y11  1.06  0.82  1.30
## 9     >=60_Y1  1.17  1.07  1.27
## 10   >=60_Y11  1.10  0.87  1.33
## 11 Diff_18-40  0.12 -0.04  0.28
## 12   Diff_<10  0.05 -0.18  0.28
## 13 Diff_11-17 -0.01 -0.24  0.22
## 14 Diff_41-59  0.18 -0.08  0.44
## 15  Diff_>=60  0.07 -0.18  0.32
```

```r
exp_interaction_CI_age(response_var = "avgProcessedokaj", design = dat.design)
```

```
##         Group Beta Lower Upper
## 1    18-40_Y1 0.54  0.42  0.66
## 2   18-40_Y11 0.51  0.27  0.75
## 3      <10_Y1 0.59  0.44  0.74
```

```
## 4     <10_Y11 0.52  0.16  0.88
## 5    11-17_Y1 0.64  0.47  0.81
## 6   11-17_Y11 0.61  0.23  0.99
## 7    41-59_Y1 0.53  0.36  0.70
## 8   41-59_Y11 0.47  0.07  0.87
## 9      >=60_Y1 0.48  0.29  0.67
## 10    >=60_Y11 0.45  0.05  0.85
## 11 Diff_18-40 0.03 -0.24  0.30
## 12   Diff_<10 0.07 -0.32  0.46
## 13 Diff_11-17 0.03 -0.39  0.45
## 14 Diff_41-59 0.06 -0.38  0.50
## 15  Diff_>=60 0.03 -0.41  0.47
```

```
exp_interaction_CI_age(response_var = "avgRedokaj", design = dat.design)
```

```
##          Group  Beta Lower Upper
## 1     18-40_Y1  0.39  0.27  0.51
## 2    18-40_Y11  0.29  0.05  0.53
## 3       <10_Y1  0.32  0.17  0.47
## 4      <10_Y11  0.29 -0.06  0.64
## 5     11-17_Y1  0.37  0.19  0.55
## 6    11-17_Y11  0.38 -0.01  0.77
## 7     41-59_Y1  0.51  0.32  0.70
## 8    41-59_Y11  0.31 -0.09  0.71
## 9      >=60_Y1  0.52  0.35  0.69
## 10    >=60_Y11  0.41  0.03  0.79
## 11 Diff_18-40  0.10 -0.17  0.37
## 12   Diff_<10  0.03 -0.35  0.41
## 13 Diff_11-17 -0.01 -0.44  0.42
## 14 Diff_41-59  0.20 -0.24  0.64
## 15  Diff_>=60  0.11 -0.31  0.53
```

```
exp_interaction_CI_age(response_var = "avgWhiteokaj", design = dat.design)
```

```
##          Group  Beta Lower Upper
## 1     18-40_Y1  0.47  0.37  0.57
## 2    18-40_Y11  0.52  0.33  0.71
## 3       <10_Y1  0.41  0.27  0.55
## 4      <10_Y11  0.49  0.18  0.80
## 5     11-17_Y1  0.47  0.34  0.60
## 6    11-17_Y11  0.53  0.23  0.83
## 7     41-59_Y1  0.36  0.17  0.55
## 8    41-59_Y11  0.42  0.05  0.79
## 9      >=60_Y1  0.33  0.11  0.55
## 10    >=60_Y11  0.35 -0.05  0.75
## 11 Diff_18-40 -0.05 -0.26  0.16
## 12   Diff_<10 -0.08 -0.42  0.26
## 13 Diff_11-17 -0.06 -0.39  0.27
## 14 Diff_41-59 -0.06 -0.48  0.36
## 15  Diff_>=60 -0.02 -0.47  0.43
```

```
#portion size
glm_interaction_CI_age(response_var = "gperokajMeat", design = dat.design)
```

```
##          Group     Beta  Lower  Upper
## 1     18-40_Y1 94.06721  86.19 101.95
```

```
## 2   18-40_Y11 85.52466  71.95  99.10
## 3       <10_Y1 49.23442  41.29  57.18
## 4      <10_Y11 46.23813  28.26  64.21
## 5     11-17_Y1 77.55558  68.57  86.54
## 6    11-17_Y11 70.66245  51.96  89.37
## 7     41-59_Y1 94.70470  84.15 105.25
## 8    41-59_Y11 85.95871  65.34 106.57
## 9      >=60_Y1 87.36548  78.46  96.27
## 10    >=60_Y11 72.06583  52.46  91.67
## 11 Diff_18-40  8.54000  -7.15  24.24
## 12    Diff_<10  3.00000 -16.65  22.65
## 13 Diff_11-17  6.89000 -13.86  27.64
## 14 Diff_41-59  8.75000 -14.41  31.90
## 15  Diff_>=60 15.30000  -6.23  36.83
```

```
glm_interaction_CI_age(response_var = "gperokajProcessed", design = dat.design)
```

```
##          Group     Beta  Lower Upper
## 1     18-40_Y1 67.23074  59.22 75.24
## 2    18-40_Y11 56.59161  43.41 69.77
## 3       <10_Y1 46.77233  38.16 55.39
## 4      <10_Y11 40.96272  23.29 58.64
## 5     11-17_Y1 62.33986  52.55 72.13
## 6    11-17_Y11 54.33392  35.01 73.66
## 7     41-59_Y1 70.97262  57.57 84.37
## 8    41-59_Y11 57.01272  32.98 81.05
## 9      >=60_Y1 57.91912  47.68 68.15
## 10    >=60_Y11 48.88804  28.85 68.93
## 11 Diff_18-40 10.64000  -4.79 26.07
## 12    Diff_<10  5.81000 -13.85 25.47
## 13 Diff_11-17  8.01000 -13.66 29.67
## 14 Diff_41-59 13.96000 -13.56 41.48
## 15  Diff_>=60  9.03000 -13.47 31.53
```

```
glm_interaction_CI_age(response_var = "gperokajRed", design = dat.design)
```

```
##          Group      Beta  Lower  Upper
## 1     18-40_Y1 103.07422  86.02 120.13
## 2    18-40_Y11  74.21319  47.15 101.27
## 3       <10_Y1  43.47392  25.55  61.39
## 4      <10_Y11  37.83113   2.17  73.49
## 5     11-17_Y1  71.85872  53.66  90.06
## 6    11-17_Y11  67.04683  30.61 103.48
## 7     41-59_Y1  95.33118  75.31 115.36
## 8    41-59_Y11  83.69360  44.38 123.01
## 9      >=60_Y1  93.43446  74.64 112.23
## 10    >=60_Y11  67.79029  30.93 104.65
## 11 Diff_18-40  28.86000  -3.12  60.85
## 12    Diff_<10   5.64000 -34.27  45.55
## 13 Diff_11-17   4.81000 -35.92  45.54
## 14 Diff_41-59  11.64000 -32.48  55.76
## 15  Diff_>=60  25.64000 -15.73  67.02
```

```
glm_interaction_CI_age(response_var = "gperokajWhite", design = dat.design)
```

```
##          Group     Beta  Lower  Upper
```

```
## 1    18-40_Y1 91.62635  80.16 103.10
## 2   18-40_Y11 89.60618  69.66 109.56
## 3      <10_Y1 50.07938  38.78  61.37
## 4     <10_Y11 49.78529  23.48  76.09
## 5    11-17_Y1 79.93022  67.68  92.18
## 6   11-17_Y11 67.94178  40.79  95.09
## 7    41-59_Y1 94.10589  78.87 109.34
## 8   41-59_Y11 88.67328  58.76 118.59
## 9     >=60_Y1 84.95860  70.51  99.41
## 10   >=60_Y11 78.13514  47.77 108.50
## 11 Diff_18-40  2.02000 -20.99  25.03
## 12   Diff_<10  0.29000 -28.33  28.92
## 13 Diff_11-17 11.99000 -17.79  41.77
## 14 Diff_41-59  5.43000 -28.14  39.00
## 15  Diff_>=60  6.82000 -26.81  40.45
```

```r
#set survey year to numeric for p-values
dat$SurveyYear <- as.numeric(dat$SurveyYear)
dat$fpc <- 15332
dat.design <-
  svydesign(
    id = ~area,
    strata = ~astrata5,
    data = dat,
    weights = ~wti,
    fpc = ~fpc
  )

#p values
summary(svyglm(MeatDays ~ SurveyYear + AgeG + SurveyYear * AgeG, family = poisson(link = "log"), dat.de
```

```
##
## Call:
## svyglm(formula = MeatDays ~ SurveyYear + AgeG + SurveyYear *
##     AgeG, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       1.1830376  0.0176420  67.058   <2e-16 ***
## SurveyYear       -0.0048489  0.0029341  -1.653   0.0986 .
## AgeG1             0.0095857  0.0212541   0.451   0.6520
## AgeG2             0.0236950  0.0226594   1.046   0.2958
## AgeG4            -0.0081174  0.0248425  -0.327   0.7439
## AgeG5            -0.0155672  0.0238140  -0.654   0.5134
## SurveyYear:AgeG1  0.0017295  0.0034185   0.506   0.6130
## SurveyYear:AgeG2  0.0044971  0.0036260   1.240   0.2150
## SurveyYear:AgeG4 -0.0024026  0.0041442  -0.580   0.5622
## SurveyYear:AgeG5 -0.0007033  0.0038181  -0.184   0.8539
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for poisson family taken to be 0.3815556)
##
## Number of Fisher Scoring iterations: 4
```

```
summary(svyglm(ProcessedDays ~ SurveyYear + AgeG + SurveyYear * AgeG, family = poisson(link = "log"), da
```

```
##
## Call:
## svyglm(formula = ProcessedDays ~ SurveyYear + AgeG + SurveyYear *
##     AgeG, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.600358   0.037152  16.160   <2e-16 ***
## SurveyYear       -0.008692   0.005909  -1.471   0.1415
## AgeG1             0.074562   0.042876   1.739   0.0822 .
## AgeG2             0.058545   0.048632   1.204   0.2288
## AgeG4            -0.066128   0.053845  -1.228   0.2196
## AgeG5            -0.087644   0.055479  -1.580   0.1143
## SurveyYear:AgeG1 -0.001705   0.006931  -0.246   0.8057
## SurveyYear:AgeG2  0.004295   0.007839   0.548   0.5838
## SurveyYear:AgeG4 -0.004203   0.008510  -0.494   0.6214
## SurveyYear:AgeG5  0.001059   0.008473   0.125   0.9005
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.9469905)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(RedDays ~ SurveyYear + AgeG + SurveyYear * AgeG, family = poisson(link = "log"), dat.des
```

```
##
## Call:
## svyglm(formula = RedDays ~ SurveyYear + AgeG + SurveyYear * AgeG,
##     design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.390655   0.039480   9.895  < 2e-16 ***
## SurveyYear       -0.013902   0.006269  -2.217  0.02672 *
## AgeG1            -0.257397   0.048653  -5.290 1.37e-07 ***
## AgeG2            -0.164415   0.056484  -2.911  0.00365 **
## AgeG4             0.096424   0.053894   1.789  0.07377 .
## AgeG5             0.206572   0.051844   3.985 7.04e-05 ***
## SurveyYear:AgeG1  0.012581   0.007433   1.693  0.09070 .
## SurveyYear:AgeG2  0.018995   0.008721   2.178  0.02953 *
```

```
## SurveyYear:AgeG4 -0.011257   0.008566  -1.314  0.18896
## SurveyYear:AgeG5 -0.008635   0.008322  -1.038  0.29960
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.8754919)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(WhiteDays ~ SurveyYear + AgeG + SurveyYear * AgeG, family = poisson(link = "log"), dat.de
```

```
##
## Call:
## svyglm(formula = WhiteDays ~ SurveyYear + AgeG + SurveyYear *
##     AgeG, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.467542   0.037753  12.384  < 2e-16 ***
## SurveyYear         0.013190   0.005640   2.339 0.019466 *
## AgeG1             -0.118295   0.048793  -2.424 0.015435 *
## AgeG2              0.057051   0.047180   1.209 0.226742
## AgeG4             -0.207264   0.054631  -3.794 0.000153 ***
## AgeG5             -0.411402   0.062941  -6.536 8.26e-11 ***
## SurveyYear:AgeG1   0.001494   0.007047   0.212 0.832132
## SurveyYear:AgeG2  -0.001092   0.007301  -0.150 0.881161
## SurveyYear:AgeG4   0.004534   0.008130   0.558 0.577161
## SurveyYear:AgeG5   0.004941   0.009081   0.544 0.586465
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.8641156)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(NoMeatDays ~ SurveyYear + AgeG + SurveyYear * AgeG, family = poisson(link = "log"), dat.d
```

```
##
## Call:
## svyglm(formula = NoMeatDays ~ SurveyYear + AgeG + SurveyYear *
##     AgeG, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -0.3012585  0.0719752  -4.186 2.99e-05 ***
## SurveyYear         0.0186511  0.0111839   1.668   0.0956 .
## AgeG1             -0.0466093  0.0891783  -0.523   0.6013
```

```
## AgeG2             -0.1180756  0.1025256  -1.152   0.2496
## AgeG4              0.0408744  0.0975489   0.419   0.6753
## AgeG5              0.0671682  0.0936281   0.717   0.4732
## SurveyYear:AgeG1 -0.0054176  0.0135106  -0.401   0.6885
## SurveyYear:AgeG2 -0.0168848  0.0157550  -1.072   0.2840
## SurveyYear:AgeG4  0.0065062  0.0150261   0.433   0.6651
## SurveyYear:AgeG5  0.0008056  0.0140577   0.057   0.9543
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1.446884)
##
## Number of Fisher Scoring iterations: 5
```

```r
summary(svyglm(avgMeatokaj ~ SurveyYear + AgeG + SurveyYear * AgeG, family = poisson(link = "log"), dat
```

```
##
## Call:
## svyglm(formula = avgMeatokaj ~ SurveyYear + AgeG + SurveyYear *
##     AgeG, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.2354255  0.0263491   8.935   <2e-16 ***
## SurveyYear       -0.0032126  0.0041476  -0.775   0.4387
## AgeG1            -0.0264709  0.0300629  -0.881   0.3787
## AgeG2             0.0357561  0.0332427   1.076   0.2823
## AgeG4            -0.0509059  0.0359335  -1.417   0.1568
## AgeG5            -0.0916455  0.0357998  -2.560   0.0106 *
## SurveyYear:AgeG1  0.0006849  0.0048221   0.142   0.8871
## SurveyYear:AgeG2  0.0040997  0.0052208   0.785   0.4324
## SurveyYear:AgeG4 -0.0058757  0.0058167  -1.010   0.3126
## SurveyYear:AgeG5 -0.0025999  0.0055643  -0.467   0.6404
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.2807978)
##
## Number of Fisher Scoring iterations: 4
```

```r
summary(svyglm(avgProcessedokaj ~ SurveyYear + AgeG + SurveyYear * AgeG, family = poisson(link = "log")
```

```
##
## Call:
## svyglm(formula = avgProcessedokaj ~ SurveyYear + AgeG + SurveyYear *
##     AgeG, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
```

```
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -0.563889   0.047129 -11.965   <2e-16 ***
## SurveyYear       -0.007646   0.007112  -1.075   0.2825
## AgeG1             0.077736   0.053239   1.460   0.1444
## AgeG2             0.081442   0.061365   1.327   0.1846
## AgeG4            -0.089087   0.062863  -1.417   0.1566
## AgeG5            -0.161419   0.065568  -2.462   0.0139 *
## SurveyYear:AgeG1 -0.004567   0.008264  -0.553   0.5805
## SurveyYear:AgeG2  0.002418   0.009408   0.257   0.7972
## SurveyYear:AgeG4 -0.005770   0.009802  -0.589   0.5562
## SurveyYear:AgeG5  0.001586   0.009847   0.161   0.8720
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3753536)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(avgRedokaj ~ SurveyYear + AgeG + SurveyYear * AgeG, family = poisson(link = "log"), dat.d
```

```
##
## Call:
## svyglm(formula = avgRedokaj ~ SurveyYear + AgeG + SurveyYear *
##     AgeG, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -0.911811   0.042286 -21.563  < 2e-16 ***
## SurveyYear       -0.010775   0.006679  -1.613   0.1069
## AgeG1            -0.269657   0.052404  -5.146 2.97e-07 ***
## AgeG2            -0.142160   0.060314  -2.357   0.0185 *
## AgeG4             0.158571   0.061549   2.576   0.0101 *
## AgeG5             0.246461   0.056988   4.325 1.61e-05 ***
## SurveyYear:AgeG1  0.009968   0.008012   1.244   0.2136
## SurveyYear:AgeG2  0.015419   0.009360   1.647   0.0997 .
## SurveyYear:AgeG4 -0.020741   0.009636  -2.153   0.0315 *
## SurveyYear:AgeG5 -0.011739   0.009065  -1.295   0.1955
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.2924239)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(avgWhiteokaj ~ SurveyYear + AgeG + SurveyYear * AgeG, family = poisson(link = "log"), dat
```

```
##
## Call:
## svyglm(formula = avgWhiteokaj ~ SurveyYear + AgeG + SurveyYear *
##     AgeG, design = dat.design, family = poisson(link = "log"))
```

```
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -0.776095   0.044596 -17.403  < 2e-16 ***
## SurveyYear          0.015978   0.006618   2.414 0.015873 *
## AgeG1              -0.150076   0.056926  -2.636 0.008455 **
## AgeG2               0.046554   0.054942   0.847 0.396932
## AgeG4              -0.240474   0.064725  -3.715 0.000209 ***
## AgeG5              -0.433748   0.074343  -5.834 6.42e-09 ***
## SurveyYear:AgeG1    0.001362   0.008251   0.165 0.868953
## SurveyYear:AgeG2   -0.001206   0.008394  -0.144 0.885743
## SurveyYear:AgeG4    0.005385   0.009575   0.562 0.573914
## SurveyYear:AgeG5   -0.001204   0.010655  -0.113 0.910042
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3197748)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(gperokajMeat ~ SurveyYear + AgeG + SurveyYear * AgeG, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajMeat ~ SurveyYear + AgeG + SurveyYear *
##     AgeG, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         92.41795    2.24935  41.086  < 2e-16 ***
## SurveyYear          -0.50413    0.33862  -1.489   0.1367
## AgeG1              -43.33534    2.38617 -18.161  < 2e-16 ***
## AgeG2              -16.87653    2.63258  -6.411 1.86e-10 ***
## AgeG4                1.62038    3.06653   0.528   0.5973
## AgeG5               -5.93054    2.80635  -2.113   0.0347 *
## SurveyYear:AgeG1     0.35258    0.35984   0.980   0.3273
## SurveyYear:AgeG2    -0.04739    0.39499  -0.120   0.9045
## SurveyYear:AgeG4    -0.36774    0.44371  -0.829   0.4073
## SurveyYear:AgeG5    -0.48162    0.43771  -1.100   0.2713
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1191.303)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(gperokajProcessed ~ SurveyYear + AgeG + SurveyYear * AgeG, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajProcessed ~ SurveyYear + AgeG + SurveyYear *
##     AgeG, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        67.50095    2.38603  28.290  < 2e-16 ***
## SurveyYear         -1.25305    0.34910  -3.589 0.000341 ***
## AgeG1             -19.80029    2.61788  -7.563 6.33e-14 ***
## AgeG2              -4.99508    3.11738  -1.602 0.109264
## AgeG4               4.73112    3.85636   1.227 0.220051
## AgeG5              -3.87798    3.38672  -1.145 0.252344
## SurveyYear:AgeG1    0.60912    0.38250   1.592 0.111458
## SurveyYear:AgeG2    0.22968    0.44783   0.513 0.608102
## SurveyYear:AgeG4   -0.24791    0.55087  -0.450 0.652740
## SurveyYear:AgeG5    0.02373    0.48100   0.049 0.960664
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1425.771)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(gperokajRed ~ SurveyYear + AgeG + SurveyYear * AgeG, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajRed ~ SurveyYear + AgeG + SurveyYear *
##     AgeG, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        94.8151     4.6221  20.513  < 2e-16 ***
## SurveyYear         -1.6742     0.6755  -2.478   0.0133 *
## AgeG1             -51.9226     4.9008 -10.595  < 2e-16 ***
## AgeG2             -24.1299     5.3099  -4.544 5.89e-06 ***
## AgeG4              -2.0875     5.5962  -0.373   0.7092
## AgeG5              -4.5545     5.3855  -0.846   0.3978
## SurveyYear:AgeG1    1.3649     0.7131   1.914   0.0558 .
## SurveyYear:AgeG2    0.7591     0.7823   0.970   0.3320
## SurveyYear:AgeG4    0.6000     0.8385   0.716   0.4744
## SurveyYear:AgeG5    0.1395     0.7994   0.175   0.8615
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2802.872)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(gperokajWhite ~ SurveyYear + AgeG + SurveyYear * AgeG, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajWhite ~ SurveyYear + AgeG + SurveyYear *
##     AgeG, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       91.30406    3.25172  28.079  < 2e-16 ***
## SurveyYear         0.00319    0.48614   0.007 0.994766
## AgeG1            -42.66962    3.34850 -12.743  < 2e-16 ***
## AgeG2            -14.05840    3.68037  -3.820 0.000138 ***
## AgeG4              5.59960    4.93497   1.135 0.256667
## AgeG5             -6.41931    4.51827  -1.421 0.155570
## SurveyYear:AgeG1   0.09133    0.51069   0.179 0.858090
## SurveyYear:AgeG2  -0.56326    0.56121  -1.004 0.315682
## SurveyYear:AgeG4  -0.75876    0.68861  -1.102 0.270670
## SurveyYear:AgeG5  -0.15153    0.68634  -0.221 0.825289
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2359.617)
##
## Number of Fisher Scoring iterations: 2
```

SI TABLE 2c

change in meat consumption behaviours by equivalised household income

```
#set survey year as factor for regression analysis
dat$SurveyYear <- as.factor(dat$SurveyYear)
dat$fpc <- 15332
dat.design <-
  svydesign(
    id = ~area,
    strata = ~astrata5,
    data = dat,
    weights = ~wti,
    fpc = ~fpc
  )

#create summary functions for exponentiation and glm regression analyses
exp_interaction_CI_eqv <- function(response_var, design) {
  model_formula <- as.formula(paste(response_var, "~ SurveyYear + eqv + SurveyYear * eqv"))
  model <- svyglm(model_formula, family = poisson(link = "log"), design = design)
  model_summary <- summary(model)
  betas <- coef(model)
  se <- coef(model_summary)[, "Std. Error"]

  # combinations of coefficients
  b_combinations <- c(
    betas["(Intercept)"],
    betas["(Intercept)"] + betas["eqv2"],
    betas["(Intercept)"] + betas["eqv3"],
    betas["(Intercept)"] + betas["SurveyYear11"],
    betas["(Intercept)"] + betas["SurveyYear11"] + betas["eqv2"] + betas["SurveyYear11:eqv2"],
    betas["(Intercept)"] + betas["SurveyYear11"] + betas["eqv3"] + betas["SurveyYear11:eqv3"]
  )

  # exponentiated coefficients
  exp_coef <- round(exp(b_combinations), 2)

  # standard errors for combinations
  se_combinations <- c(
    se["(Intercept)"],
    sqrt(se["(Intercept)"]^2 + se["eqv2"]^2),
    sqrt(se["(Intercept)"]^2 + se["eqv3"]^2),
    sqrt(se["(Intercept)"]^2 + se["SurveyYear11"]^2),
    sqrt(se["(Intercept)"]^2 + se["SurveyYear11"]^2 + se["eqv2"]^2 + se["SurveyYear11:eqv2"]^2),
    sqrt(se["(Intercept)"]^2 + se["SurveyYear11"]^2 + se["eqv3"]^2 + se["SurveyYear11:eqv3"]^2)
  )

  # CI for combinations
  lower_bound <- round(exp_coef - 1.96 * se_combinations, 2)
  upper_bound <- round(exp_coef + 1.96 * se_combinations, 2)

  # differences of exponentiated coefficients for each pair of groups (Y1 - Y11)
  diff_coefs <- exp_coef[c(1, 2, 3)] - exp_coef[c(4, 5, 6)]
```

```
  # standard errors for the differences
  se_diff <- sqrt(se_combinations[c(1, 2, 3)]^2 + se_combinations[c(4, 5, 6)]^2)

  # CI for the differences
  diff_lower_bound <- round(diff_coefs - 1.96 * se_diff, 2)
  diff_upper_bound <- round(diff_coefs + 1.96 * se_diff, 2)

  # data frame to present the results
  result_table <- rbind(
    data.frame(
      Group = c("eqv1_Y1", "eqv2_Y1", "eqv3_Y1", "eqv1_Y11", "eqv2_Y11", "eqv3_Y11"),
      Beta = exp_coef,
      Lower = lower_bound,
      Upper = upper_bound
    ),
    data.frame(
      Group = c("Diff_eqv1", "Diff_eqv2", "Diff_eqv3"),
      Beta = round(diff_coefs, 2),
      Lower = round(diff_lower_bound, 2),
      Upper = round(diff_upper_bound, 2)
    )
  )
  return(result_table)
}
glm_interaction_CI_eqv <- function(response_var, design) {
  model_formula <- as.formula(paste(response_var, "~ SurveyYear + eqv + SurveyYear * eqv"))
  model <- svyglm(model_formula, design = design)
  model_summary <- summary(model)
  betas <- coef(model)
  se <- coef(model_summary)[, "Std. Error"]

  # combinations of coefficients
  b_combinations <- c(
    betas["(Intercept)"],
    betas["(Intercept)"] + betas["eqv2"],
    betas["(Intercept)"] + betas["eqv3"],
    betas["(Intercept)"] + betas["SurveyYear11"],
    betas["(Intercept)"] + betas["SurveyYear11"] + betas["eqv2"] + betas["SurveyYear11:eqv2"],
    betas["(Intercept)"] + betas["SurveyYear11"] + betas["eqv3"] + betas["SurveyYear11:eqv3"]
  )

  # standard errors for combinations
  se_combinations <- c(
    se["(Intercept)"],
    sqrt(se["(Intercept)"]^2 + se["eqv2"]^2),
    sqrt(se["(Intercept)"]^2 + se["eqv3"]^2),
    sqrt(se["(Intercept)"]^2 + se["SurveyYear11"]^2),
    sqrt(se["(Intercept)"]^2 + se["SurveyYear11"]^2 + se["eqv2"]^2 + se["SurveyYear11:eqv2"]^2),
    sqrt(se["(Intercept)"]^2 + se["SurveyYear11"]^2 + se["eqv3"]^2 + se["SurveyYear11:eqv3"]^2)
  )

  # CI for combinations
  lower_bound <- round(b_combinations - 1.96 * se_combinations, 2)
```

```r
  upper_bound <- round(b_combinations + 1.96 * se_combinations, 2)

  # differences of coefficients for each pair of groups (Y1 - Y11)
  diff_coefs <- b_combinations[c(1, 2, 3)] - b_combinations[c(4, 5, 6)]

  # standard errors for the differences
  se_diff <- sqrt(se_combinations[c(1, 2, 3)]^2 + se_combinations[c(4, 5, 6)]^2)

  # CI for the differences
  diff_lower_bound <- round(diff_coefs - 1.96 * se_diff, 2)
  diff_upper_bound <- round(diff_coefs + 1.96 * se_diff, 2)

  # data frame to present the results
  result_table <- rbind(
    data.frame(
      Group = c("eqv1_Y1", "eqv2_Y1", "eqv3_Y1", "eqv1_Y11", "eqv2_Y11", "eqv3_Y11"),
      Beta = b_combinations,
      Lower = lower_bound,
      Upper = upper_bound
    ),
    data.frame(
      Group = c("Diff_eqv1", "Diff_eqv2", "Diff_eqv3"),
      Beta = round(diff_coefs, 2),
      Lower = round(diff_lower_bound, 2),
      Upper = round(diff_upper_bound, 2)
    )
  )

  return(result_table)
}

#days
exp_interaction_CI_eqv(response_var = "MeatDays", design = dat.design)
```

```
##       Group Beta Lower Upper
## 1   eqv1_Y1 3.26  3.21  3.31
## 2   eqv2_Y1 3.38  3.30  3.46
## 3   eqv3_Y1 3.31  3.24  3.38
## 4  eqv1_Y11 3.12  3.03  3.21
## 5  eqv2_Y11 3.16  3.01  3.31
## 6  eqv3_Y11 3.01  2.87  3.15
## 7 Diff_eqv1 0.14  0.04  0.24
## 8 Diff_eqv2 0.22  0.05  0.39
## 9 Diff_eqv3 0.30  0.14  0.46
```

```r
exp_interaction_CI_eqv(response_var = "ProcessedDays", design = dat.design)
```

```
##       Group Beta Lower Upper
## 1   eqv1_Y1 1.79  1.69  1.89
## 2   eqv2_Y1 1.84  1.66  2.02
## 3   eqv3_Y1 1.80  1.63  1.97
## 4  eqv1_Y11 1.56  1.38  1.74
## 5  eqv2_Y11 1.80  1.48  2.12
## 6  eqv3_Y11 1.63  1.33  1.93
```

```
## 7 Diff_eqv1 0.23  0.02  0.44
## 8 Diff_eqv2 0.04 -0.33  0.41
## 9 Diff_eqv3 0.17 -0.18  0.52
```

```r
exp_interaction_CI_eqv(response_var = "RedDays", design = dat.design)
```

```
##       Group Beta Lower Upper
## 1   eqv1_Y1 1.58  1.48  1.68
## 2   eqv2_Y1 1.56  1.37  1.75
## 3   eqv3_Y1 1.59  1.42  1.76
## 4  eqv1_Y11 1.27  1.08  1.46
## 5  eqv2_Y11 1.32  0.98  1.66
## 6  eqv3_Y11 1.15  0.82  1.48
## 7 Diff_eqv1 0.31  0.09  0.53
## 8 Diff_eqv2 0.24 -0.15  0.63
## 9 Diff_eqv3 0.44  0.07  0.81
```

```r
exp_interaction_CI_eqv(response_var = "WhiteDays", design = dat.design)
```

```
##       Group  Beta Lower Upper
## 1   eqv1_Y1  1.53  1.43  1.63
## 2   eqv2_Y1  1.36  1.16  1.56
## 3   eqv3_Y1  1.47  1.29  1.65
## 4  eqv1_Y11  1.52  1.33  1.71
## 5  eqv2_Y11  1.65  1.31  1.99
## 6  eqv3_Y11  1.62  1.30  1.94
## 7 Diff_eqv1  0.01 -0.20  0.22
## 8 Diff_eqv2 -0.29 -0.69  0.11
## 9 Diff_eqv3 -0.15 -0.52  0.22
```

```r
exp_interaction_CI_eqv(response_var = "NoMeatDays", design = dat.design)
```

```
##       Group  Beta Lower Upper
## 1   eqv1_Y1  0.74  0.53  0.95
## 2   eqv2_Y1  0.62  0.25  0.99
## 3   eqv3_Y1  0.69  0.36  1.02
## 4  eqv1_Y11  0.88  0.52  1.24
## 5  eqv2_Y11  0.84  0.20  1.48
## 6  eqv3_Y11  0.99  0.43  1.55
## 7 Diff_eqv1 -0.14 -0.56  0.28
## 8 Diff_eqv2 -0.22 -0.96  0.52
## 9 Diff_eqv3 -0.30 -0.95  0.35
```

```r
#occasions
exp_interaction_CI_eqv(response_var = "avgMeatokaj", design = dat.design)
```

```
##       Group Beta Lower Upper
## 1   eqv1_Y1 1.29  1.22  1.36
## 2   eqv2_Y1 1.24  1.12  1.36
## 3   eqv3_Y1 1.26  1.15  1.37
## 4  eqv1_Y11 1.15  1.03  1.27
## 5  eqv2_Y11 1.23  1.01  1.45
## 6  eqv3_Y11 1.11  0.91  1.31
## 7 Diff_eqv1 0.14  0.00  0.28
## 8 Diff_eqv2 0.01 -0.24  0.26
## 9 Diff_eqv3 0.15 -0.08  0.38
```

```r
exp_interaction_CI_eqv(response_var = "avgProcessedokaj", design = dat.design)
```

```
##       Group  Beta Lower Upper
## 1   eqv1_Y1  0.56  0.43  0.69
## 2   eqv2_Y1  0.54  0.33  0.75
## 3   eqv3_Y1  0.56  0.35  0.77
## 4  eqv1_Y11  0.49  0.27  0.71
## 5  eqv2_Y11  0.56  0.18  0.94
## 6  eqv3_Y11  0.50  0.13  0.87
## 7 Diff_eqv1  0.07 -0.19  0.33
## 8 Diff_eqv2 -0.02 -0.45  0.41
## 9 Diff_eqv3  0.06 -0.37  0.49
```

```r
exp_interaction_CI_eqv(response_var = "avgRedokaj", design = dat.design)
```

```
##       Group Beta Lower Upper
## 1   eqv1_Y1 0.45  0.34  0.56
## 2   eqv2_Y1 0.43  0.22  0.64
## 3   eqv3_Y1 0.44  0.25  0.63
## 4  eqv1_Y11 0.35  0.14  0.56
## 5  eqv2_Y11 0.37 -0.01  0.75
## 6  eqv3_Y11 0.31 -0.05  0.67
## 7 Diff_eqv1 0.10 -0.14  0.34
## 8 Diff_eqv2 0.06 -0.37  0.49
## 9 Diff_eqv3 0.13 -0.28  0.54
```

```r
exp_interaction_CI_eqv(response_var = "avgWhiteokaj", design = dat.design)
```

```
##       Group  Beta Lower Upper
## 1   eqv1_Y1  0.43  0.32  0.54
## 2   eqv2_Y1  0.39  0.16  0.62
## 3   eqv3_Y1  0.41  0.22  0.60
## 4  eqv1_Y11  0.44  0.24  0.64
## 5  eqv2_Y11  0.46  0.07  0.85
## 6  eqv3_Y11  0.47  0.12  0.82
## 7 Diff_eqv1 -0.01 -0.24  0.22
## 8 Diff_eqv2 -0.07 -0.52  0.38
## 9 Diff_eqv3 -0.06 -0.46  0.34
```

```r
#portion size
glm_interaction_CI_eqv(response_var = "gperokajMeat", design = dat.design)
```

```
##       Group     Beta  Lower  Upper
## 1   eqv1_Y1 81.10645  74.01  88.20
## 2   eqv2_Y1 90.05789  78.04 102.07
## 3   eqv3_Y1 86.06618  75.01  97.12
## 4  eqv1_Y11 69.39970  58.43  80.37
## 5  eqv2_Y11 81.69669  62.62 100.77
## 6  eqv3_Y11 78.65233  60.90  96.41
## 7 Diff_eqv1 11.71000  -1.36  24.77
## 8 Diff_eqv2  8.36000 -14.18  30.90
## 9 Diff_eqv3  7.41000 -13.50  28.33
```

```r
glm_interaction_CI_eqv(response_var = "gperokajProcessed", design = dat.design)
```

```
##       Group     Beta  Lower Upper
## 1   eqv1_Y1 65.70271  56.08 75.32
```

```
## 2   eqv2_Y1 66.25288  51.31 81.19
## 3   eqv3_Y1 59.11892  44.02 74.22
## 4  eqv1_Y11 51.75540  37.22 66.29
## 5  eqv2_Y11 56.17113  32.95 79.40
## 6  eqv3_Y11 52.06074  28.45 75.67
## 7 Diff_eqv1 13.95000  -3.48 31.38
## 8 Diff_eqv2 10.08000 -17.53 37.70
## 9 Diff_eqv3  7.06000 -20.96 35.08
```

```
glm_interaction_CI_eqv(response_var = "gperokajRed", design = dat.design)
```

```
##       Group     Beta  Lower  Upper
## 1   eqv1_Y1 75.80272  67.08  84.53
## 2   eqv2_Y1 93.85080  78.22 109.49
## 3   eqv3_Y1 93.65726  78.33 108.99
## 4  eqv1_Y11 67.37933  52.03  82.73
## 5  eqv2_Y11 72.93400  46.08  99.78
## 6  eqv3_Y11 70.62557  43.66  97.59
## 7 Diff_eqv1  8.42000  -9.24  26.08
## 8 Diff_eqv2 20.92000 -10.15  51.99
## 9 Diff_eqv3 23.03000  -7.99  54.05
```

```
glm_interaction_CI_eqv(response_var = "gperokajWhite", design = dat.design)
```

```
##       Group     Beta  Lower  Upper
## 1   eqv1_Y1 78.19691  69.97  86.42
## 2   eqv2_Y1 85.89591  69.78 102.01
## 3   eqv3_Y1 90.08492  75.99 104.18
## 4  eqv1_Y11 72.56214  58.38  86.74
## 5  eqv2_Y11 87.07504  59.58 114.57
## 6  eqv3_Y11 78.27256  54.56 101.99
## 7 Diff_eqv1  5.63000 -10.76  22.03
## 8 Diff_eqv2 -1.18000 -33.04  30.69
## 9 Diff_eqv3 11.81000 -15.77  39.40
```

```
#set survey year to numeric for p-values
dat$SurveyYear <- as.numeric(dat$SurveyYear)
dat$fpc <- 15332
dat.design <-
  svydesign(
    id = ~area,
    strata = ~astrata5,
    data = dat,
    weights = ~wti,
    fpc = ~fpc
  )

#p values
summary(svyglm(MeatDays ~ SurveyYear + eqv + SurveyYear * eqv, family = poisson(link = "log"), dat.desi
```

```
##
## Call:
## svyglm(formula = MeatDays ~ SurveyYear + eqv + SurveyYear * eqv,
##     design = dat.design, family = poisson(link = "log"))
##
## Survey design:
```

```
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       1.1829088  0.0168305  70.284   <2e-16 ***
## SurveyYear       -0.0058929  0.0027522  -2.141   0.0324 *
## eqv2             -0.0085235  0.0227556  -0.375   0.7080
## eqv3              0.0076743  0.0215995   0.355   0.7224
## SurveyYear:eqv2   0.0043465  0.0036481   1.191   0.2336
## SurveyYear:eqv3  -0.0001642  0.0035300  -0.047   0.9629
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3652429)
##
## Number of Fisher Scoring iterations: 4
```

```
summary(svyglm(ProcessedDays ~ SurveyYear + eqv + SurveyYear * eqv, family = poisson(link = "log"), dat
```

```
##
## Call:
## svyglm(formula = ProcessedDays ~ SurveyYear + eqv + SurveyYear *
##     eqv, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.592510   0.036646  16.169  < 2e-16 ***
## SurveyYear       -0.015946   0.005731  -2.783  0.00545 **
## eqv2             -0.001866   0.050225  -0.037  0.97037
## eqv3             -0.013110   0.048754  -0.269  0.78803
## SurveyYear:eqv2   0.008229   0.007967   1.033  0.30179
## SurveyYear:eqv3   0.012329   0.007507   1.642  0.10070
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.9139849)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(RedDays ~ SurveyYear + eqv + SurveyYear * eqv, family = poisson(link = "log"), dat.desig
```

```
##
## Call:
## svyglm(formula = RedDays ~ SurveyYear + eqv + SurveyYear * eqv,
##     design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
```

```
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.4119271  0.0386242  10.665   <2e-16 ***
## SurveyYear      -0.0192965  0.0060306  -3.200   0.0014 **
## eqv2            -0.0004578  0.0556236  -0.008   0.9934
## eqv3             0.0390208  0.0515949   0.756   0.4496
## SurveyYear:eqv2  0.0080522  0.0086871   0.927   0.3541
## SurveyYear:eqv3  0.0040994  0.0081467   0.503   0.6149
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.8640548)
##
## Number of Fisher Scoring iterations: 5
```

```r
summary(svyglm(WhiteDays ~ SurveyYear + eqv + SurveyYear * eqv, family = poisson(link = "log"), dat.des
```

```
##
## Call:
## svyglm(formula = WhiteDays ~ SurveyYear + eqv + SurveyYear *
##     eqv, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.354727   0.038511   9.211   <2e-16 ***
## SurveyYear       0.008767   0.005905   1.485   0.1378
## eqv2            -0.087504   0.058372  -1.499   0.1340
## eqv3            -0.002378   0.050912  -0.047   0.9627
## SurveyYear:eqv2  0.017322   0.008461   2.047   0.0408 *
## SurveyYear:eqv3  0.003543   0.007602   0.466   0.6413
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.8596569)
##
## Number of Fisher Scoring iterations: 5
```

```r
summary(svyglm(NoMeatDays ~ SurveyYear + eqv + SurveyYear * eqv, family = poisson(link = "log"), dat.de
```

```
##
## Call:
## svyglm(formula = NoMeatDays ~ SurveyYear + eqv + SurveyYear *
##     eqv, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -0.298130   0.067474  -4.418 1.06e-05 ***
```

```
## SurveyYear       0.022002   0.010189   2.159    0.031 *
## eqv2             0.029347   0.092456   0.317    0.751
## eqv3            -0.033695   0.087919  -0.383    0.702
## SurveyYear:eqv2 -0.015747   0.014053  -1.121    0.263
## SurveyYear:eqv3  0.001353   0.013238   0.102    0.919
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1.419507)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(avgMeatokaj ~ SurveyYear + eqv + SurveyYear * eqv, family = poisson(link = "log"), dat.de
```

```
##
## Call:
## svyglm(formula = avgMeatokaj ~ SurveyYear + eqv + SurveyYear *
##     eqv, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.218815   0.023948   9.137   <2e-16 ***
## SurveyYear      -0.009302   0.003737  -2.489   0.0129 *
## eqv2            -0.027191   0.033885  -0.802   0.4224
## eqv3            -0.017212   0.031480  -0.547   0.5846
## SurveyYear:eqv2  0.009273   0.005267   1.761   0.0785 .
## SurveyYear:eqv3  0.006206   0.004895   1.268   0.2050
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.2745977)
##
## Number of Fisher Scoring iterations: 4
```

```
summary(svyglm(avgProcessedokaj ~ SurveyYear + eqv + SurveyYear * eqv, family = poisson(link = "log"), d
```

```
##
## Call:
## svyglm(formula = avgProcessedokaj ~ SurveyYear + eqv + SurveyYear *
##     eqv, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -0.586401   0.043556 -13.463   <2e-16 ***
## SurveyYear      -0.016077   0.006722  -2.392   0.0169 *
## eqv2            -0.010910   0.058194  -0.187   0.8513
## eqv3            -0.013967   0.057126  -0.244   0.8069
```

```
## SurveyYear:eqv2  0.010300   0.009196   1.120   0.2628
## SurveyYear:eqv3  0.012366   0.008768   1.410   0.1586
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3689228)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(avgRedokaj ~ SurveyYear + eqv + SurveyYear * eqv, family = poisson(link = "log"), dat.de
```

```
##
## Call:
## svyglm(formula = avgRedokaj ~ SurveyYear + eqv + SurveyYear *
##     eqv, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -0.866223   0.041590 -20.828  < 2e-16 ***
## SurveyYear       -0.020162   0.006468  -3.117  0.00186 **
## eqv2             -0.006231   0.060610  -0.103  0.91813
## eqv3              0.037544   0.056129   0.669  0.50366
## SurveyYear:eqv2  0.009169   0.009381   0.977  0.32846
## SurveyYear:eqv3  0.004931   0.008776   0.562  0.57429
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.2837342)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(avgWhiteokaj ~ SurveyYear + eqv + SurveyYear * eqv, family = poisson(link = "log"), dat.d
```

```
##
## Call:
## svyglm(formula = avgWhiteokaj ~ SurveyYear + eqv + SurveyYear *
##     eqv, design = dat.design, family = poisson(link = "log"))
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -0.915064   0.041298 -22.158   <2e-16 ***
## SurveyYear        0.012377   0.006481   1.910   0.0563 .
## eqv2             -0.057836   0.069583  -0.831   0.4060
## eqv3             -0.014626   0.055120  -0.265   0.7908
## SurveyYear:eqv2  0.012298   0.010039   1.225   0.2207
## SurveyYear:eqv3  0.004274   0.008394   0.509   0.6107
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 0.3223685)
##
## Number of Fisher Scoring iterations: 5
```

```
summary(svyglm(gperokajMeat ~ SurveyYear + eqv + SurveyYear * eqv, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajMeat ~ SurveyYear + eqv + SurveyYear *
##     eqv, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      79.5673     1.9993  39.797  < 2e-16 ***
## SurveyYear       -0.4245     0.2947  -1.440  0.14992
## eqv2              8.1965     2.8097   2.917  0.00358 **
## eqv3              7.3963     2.5973   2.848  0.00446 **
## SurveyYear:eqv2  -0.4364     0.4092  -1.067  0.28633
## SurveyYear:eqv3  -0.2436     0.3877  -0.628  0.52988
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1334.372)
##
## Number of Fisher Scoring iterations: 2
```

```
summary(svyglm(gperokajProcessed ~ SurveyYear + eqv + SurveyYear * eqv, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajProcessed ~ SurveyYear + eqv + SurveyYear *
##     eqv, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     66.31434    2.69294  24.625  < 2e-16 ***
## SurveyYear      -1.35663    0.36524  -3.714  0.00021 ***
## eqv2             0.71544    3.43462   0.208  0.83502
## eqv3            -3.15855    3.47272  -0.910  0.36320
## SurveyYear:eqv2  0.06088    0.47301   0.129  0.89761
## SurveyYear:eqv3  0.26282    0.48521   0.542  0.58812
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1456.8)
```

```
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(gperokajRed ~ SurveyYear + eqv + SurveyYear * eqv, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajRed ~ SurveyYear + eqv + SurveyYear *
##     eqv, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        73.3252     2.9672  24.712  < 2e-16 ***
## SurveyYear         -0.1542     0.4454  -0.346  0.72926
## eqv2               16.1207     4.2781   3.768  0.00017 ***
## eqv3               16.7655     4.1161   4.073 4.85e-05 ***
## SurveyYear:eqv2    -1.6120     0.6422  -2.510  0.01215 *
## SurveyYear:eqv3    -1.1277     0.6230  -1.810  0.07045 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2855.013)
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(svyglm(gperokajWhite ~ SurveyYear + eqv + SurveyYear * eqv, dat.design))
```

```
##
## Call:
## svyglm(formula = gperokajWhite ~ SurveyYear + eqv + SurveyYear *
##     eqv, design = dat.design)
##
## Survey design:
## svydesign(id = ~area, strata = ~astrata5, data = dat, weights = ~wti,
##     fpc = ~fpc)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        80.6169     2.6313  30.637   <2e-16 ***
## SurveyYear         -0.1762     0.4024  -0.438   0.6615
## eqv2                5.5822     4.2278   1.320   0.1869
## eqv3                7.5437     3.6644   2.059   0.0397 *
## SurveyYear:eqv2    -0.1058     0.6155  -0.172   0.8635
## SurveyYear:eqv3    -0.3017     0.5332  -0.566   0.5716
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2460.043)
##
## Number of Fisher Scoring iterations: 2
```

SI TABLE 3

change in meat consumption behaviours (Table 2), all years reported

See Table 2 section, all years are reported within those analyses.

SI TABLE 4

demographic characteristics (Table 1), all years reported

```
#specify survey weighting structure for descriptive analysis
survey_design <- dat %>%
  as_survey_design(ids = area, # cluster ids
                   weights = wti, # weight variable created above
                   strata = astrata5 # sampling was stratified by district
  )

#YEAR-BY-YEAR DEMOGRAPHIC ANALYSIS

# Creating a list of years
years <- unique(dat$SurveyYear)

# Define a function that runs your analyses on a survey design
analyze_data <- function(year) {

  # Subset data for the year
  dat_subset <- dat %>% filter(SurveyYear == year)

  # Define survey design for subset
  survey_design <- dat_subset %>%
    as_survey_design(ids = area, # cluster ids
                     weights = wti, # weight variable created above
                     strata = astrata5 # sampling was stratified by district
    )

  # age groups
  age <- table(dat_subset$AgeG)
  age_pct <- survey_design %>%
    group_by(AgeG) %>%
    summarise(pct = survey_mean(na.rm = TRUE))

  # sex
  sex <- table(dat_subset$Sex)
  sex_pct <- survey_design %>%
    group_by(Sex) %>%
    summarise(pct = survey_mean(na.rm = TRUE))

  # income tertiles
  income <- table(dat_subset$eqv)
  income_pct <- survey_design %>%
    group_by(eqv) %>%
    summarise(pct = survey_mean(na.rm = TRUE))

  # meat consumers
  survey_design <- mutate(survey_design, meat_gt_0 = as.numeric(sumMeatg > 0))
  meat <- table(survey_design$variables$meat_gt_0)
  meat_pct <- survey_design %>%
    group_by(meat_gt_0) %>%
    summarise(pct = survey_mean(na.rm = TRUE))
```

```r
  return(list(age = age, age_pct = age_pct,
              sex = sex, sex_pct = sex_pct,
              income = income, income_pct = income_pct,
              meat = meat, meat_pct = meat_pct))
}

#apply function to each year of the data
results <- map(years, ~analyze_data(.x))

#names of the results list will be the years
names(results) <- years

# Loop through the years and print results
for(year in names(results)) {
  cat("\nResults for Survey Year:", year, "\n")

  cat("\nAge groups:\n")
  print(results[[year]]$age)
  cat("\nAge groups percentage:\n")
  print(results[[year]]$age_pct)

  cat("\nSex:\n")
  print(results[[year]]$sex)
  cat("\nSex percentage:\n")
  print(results[[year]]$sex_pct)

  cat("\nIncome tertiles:\n")
  print(results[[year]]$income)
  cat("\nIncome tertiles percentage:\n")
  print(results[[year]]$income_pct)

  cat("\nMeat consumers:\n")
  print(results[[year]]$meat)
  cat("\nMeat consumers percentage:\n")
  print(results[[year]]$meat_pct)
}
```

```
##
## Results for Survey Year: 1
##
## Age groups:
##
##   3   1   2   4   5
## 313 505 296 281 234
##
## Age groups percentage:
## # A tibble: 5 x 3
##   AgeG     pct  pct_se
##   <fct>  <dbl>   <dbl>
## 1 3     0.322  0.0181
## 2 1     0.125  0.00685
## 3 2     0.0792 0.00552
## 4 4     0.252  0.0161
## 5 5     0.222  0.0165
```

```
##
## Sex:
##
##   M   F
## 747 882
##
## Sex percentage:
## # A tibble: 2 x 3
##   Sex     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 M     0.477 0.0205
## 2 F     0.523 0.0205
##
## Income tertiles:
##
##   1   2   3
## 506 448 470
##
## Income tertiles percentage:
## # A tibble: 4 x 3
##   eqv     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 1     0.269 0.0195
## 2 2     0.273 0.0186
## 3 3     0.296 0.0192
## 4 <NA>  0.162 0.0223
##
## Meat consumers:
##
##    0    1
##   54 1575
##
## Meat consumers percentage:
## # A tibble: 2 x 3
##   meat_gt_0    pct  pct_se
##       <dbl>  <dbl>   <dbl>
## 1         0 0.0357 0.00739
## 2         1 0.964  0.00739
##
## Results for Survey Year: 2
##
## Age groups:
##
##   3   1   2   4   5
## 321 459 346 264 249
##
## Age groups percentage:
## # A tibble: 5 x 3
##   AgeG     pct  pct_se
##   <fct>  <dbl>   <dbl>
## 1 3     0.312  0.0213
## 2 1     0.104  0.00652
## 3 2     0.0985 0.00644
## 4 4     0.246  0.0162
```

```
## 5 5     0.239  0.0207
##
## Sex:
##
##   M   F
## 770 869
##
## Sex percentage:
## # A tibble: 2 x 3
##   Sex     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 M     0.496 0.0198
## 2 F     0.504 0.0198
##
## Income tertiles:
##
##   1   2   3
## 450 505 504
##
## Income tertiles percentage:
## # A tibble: 4 x 3
##   eqv     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 1     0.234 0.0201
## 2 2     0.288 0.0197
## 3 3     0.338 0.0252
## 4 <NA>  0.140 0.0189
##
## Meat consumers:
##
##    0    1
##   54 1585
##
## Meat consumers percentage:
## # A tibble: 2 x 3
##   meat_gt_0    pct  pct_se
##       <dbl>  <dbl>   <dbl>
## 1         0 0.0458 0.00887
## 2         1 0.954  0.00887
##
## Results for Survey Year: 3
##
## Age groups:
##
##   3   1   2   4   5
## 319 400 316 278 216
##
## Age groups percentage:
## # A tibble: 5 x 3
##   AgeG     pct  pct_se
##   <fct>  <dbl>   <dbl>
## 1 3     0.325  0.0194
## 2 1     0.112  0.00662
## 3 2     0.0936 0.00768
```

```
## 4 4     0.269  0.0172
## 5 5     0.200  0.0180
##
## Sex:
##
##   M   F
## 720 809
##
## Sex percentage:
## # A tibble: 2 x 3
##   Sex     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 M     0.510 0.0191
## 2 F     0.490 0.0191
##
## Income tertiles:
##
##   1   2   3
## 444 463 424
##
## Income tertiles percentage:
## # A tibble: 4 x 3
##   eqv     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 1     0.248 0.0196
## 2 2     0.305 0.0200
## 3 3     0.291 0.0212
## 4 <NA>  0.155 0.0189
##
## Meat consumers:
##
##    0    1
##   61 1468
##
## Meat consumers percentage:
## # A tibble: 2 x 3
##   meat_gt_0    pct  pct_se
##       <dbl>  <dbl>   <dbl>
## 1         0 0.0484 0.00808
## 2         1 0.952  0.00808
##
## Results for Survey Year: 4
##
## Age groups:
##
##   3   1   2   4   5
## 366 478 365 374 319
##
## Age groups percentage:
## # A tibble: 5 x 3
##   AgeG     pct  pct_se
##   <fct>  <dbl>   <dbl>
## 1 3     0.289  0.0199
## 2 1     0.102  0.00633
```

```
## 3 2     0.0866 0.00573
## 4 4     0.276  0.0182
## 5 5     0.247  0.0190
##
## Sex:
##
##    M    F
##  862 1040
##
## Sex percentage:
## # A tibble: 2 x 3
##   Sex     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 M     0.487 0.0195
## 2 F     0.513 0.0195
##
## Income tertiles:
##
##   1   2   3
## 526 530 552
##
## Income tertiles percentage:
## # A tibble: 4 x 3
##   eqv     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 1     0.233 0.0194
## 2 2     0.299 0.0192
## 3 3     0.312 0.0194
## 4 <NA>  0.156 0.0171
##
## Meat consumers:
##
##    0    1
##   83 1819
##
## Meat consumers percentage:
## # A tibble: 2 x 3
##   meat_gt_0    pct  pct_se
##       <dbl>  <dbl>   <dbl>
## 1         0 0.0507 0.00828
## 2         1 0.949  0.00828
##
## Results for Survey Year: 5
##
## Age groups:
##
##   3   1   2   4   5
## 232 318 208 209 202
##
## Age groups percentage:
## # A tibble: 5 x 3
##   AgeG     pct  pct_se
##   <fct>  <dbl>   <dbl>
## 1 3      0.307  0.0239
```

```
## 2 1     0.104  0.00713
## 3 2     0.0750 0.00671
## 4 4     0.268  0.0176
## 5 5     0.247  0.0199
##
## Sex:
##
##   M   F
## 524 645
##
## Sex percentage:
## # A tibble: 2 x 3
##   Sex     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 M     0.496 0.0200
## 2 F     0.504 0.0200
##
## Income tertiles:
##
##   1   2   3
## 367 340 333
##
## Income tertiles percentage:
## # A tibble: 4 x 3
##   eqv     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 1     0.310 0.0228
## 2 2     0.260 0.0191
## 3 3     0.310 0.0224
## 4 <NA>  0.120 0.0162
##
## Meat consumers:
##
##    0    1
##   47 1122
##
## Meat consumers percentage:
## # A tibble: 2 x 3
##   meat_gt_0    pct  pct_se
##       <dbl>  <dbl>   <dbl>
## 1         0 0.0355 0.00647
## 2         1 0.964  0.00647
##
## Results for Survey Year: 6
##
## Age groups:
##
##   3   1   2   4   5
## 244 382 273 229 203
##
## Age groups percentage:
## # A tibble: 5 x 3
##   AgeG     pct  pct_se
##   <fct>  <dbl>   <dbl>
```

```
## 1 3     0.299  0.0210
## 2 1     0.126  0.00752
## 3 2     0.0962 0.00811
## 4 4     0.263  0.0191
## 5 5     0.216  0.0172
##
## Sex:
##
##   M   F
## 598 733
##
## Sex percentage:
## # A tibble: 2 x 3
##   Sex     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 M     0.484 0.0192
## 2 F     0.516 0.0192
##
## Income tertiles:
##
##   1   2   3
## 368 397 404
##
## Income tertiles percentage:
## # A tibble: 4 x 3
##   eqv     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 1     0.256 0.0231
## 2 2     0.288 0.0199
## 3 3     0.338 0.0251
## 4 <NA>  0.118 0.0150
##
## Meat consumers:
##
##    0    1
##   49 1282
##
## Meat consumers percentage:
## # A tibble: 2 x 3
##   meat_gt_0    pct  pct_se
##       <dbl>  <dbl>   <dbl>
## 1         0 0.0480 0.00894
## 2         1 0.952  0.00894
##
## Results for Survey Year: 7
##
## Age groups:
##
##   3   1   2   4   5
## 266 370 233 234 225
##
## Age groups percentage:
## # A tibble: 5 x 3
##   AgeG     pct  pct_se
```

```
##   <fct>  <dbl>   <dbl>
## 1 3     0.313  0.0205
## 2 1     0.114  0.00710
## 3 2     0.0803 0.00635
## 4 4     0.249  0.0192
## 5 5     0.243  0.0174
##
## Sex:
##
##   M   F
## 642 686
##
## Sex percentage:
## # A tibble: 2 x 3
##   Sex     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 M     0.521 0.0201
## 2 F     0.479 0.0201
##
## Income tertiles:
##
##   1   2   3
## 396 419 355
##
## Income tertiles percentage:
## # A tibble: 4 x 3
##   eqv     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 1     0.270 0.0179
## 2 2     0.304 0.0190
## 3 3     0.286 0.0213
## 4 <NA>  0.140 0.0181
##
## Meat consumers:
##
##    0    1
##   51 1277
##
## Meat consumers percentage:
## # A tibble: 2 x 3
##   meat_gt_0    pct  pct_se
##       <dbl>  <dbl>   <dbl>
## 1         0 0.0464 0.00897
## 2         1 0.954  0.00897
##
## Results for Survey Year: 8
##
## Age groups:
##
##   3   1   2   4   5
## 251 383 229 268 208
##
## Age groups percentage:
## # A tibble: 5 x 3
```

```
##   AgeG      pct  pct_se
##   <fct>  <dbl>   <dbl>
## 1 3     0.295  0.0218
## 2 1     0.123  0.00799
## 3 2     0.0818 0.00694
## 4 4     0.275  0.0190
## 5 5     0.226  0.0193
##
## Sex:
##
##   M   F
## 607 732
##
## Sex percentage:
## # A tibble: 2 x 3
##   Sex     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 M     0.465 0.0226
## 2 F     0.535 0.0226
##
## Income tertiles:
##
##   1   2   3
## 379 360 426
##
## Income tertiles percentage:
## # A tibble: 4 x 3
##   eqv     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 1     0.267 0.0184
## 2 2     0.240 0.0205
## 3 3     0.333 0.0243
## 4 <NA>  0.160 0.0218
##
## Meat consumers:
##
##    0    1
##   53 1286
##
## Meat consumers percentage:
## # A tibble: 2 x 3
##   meat_gt_0    pct pct_se
##       <dbl>  <dbl>  <dbl>
## 1         0 0.0485 0.0106
## 2         1 0.952  0.0106
##
## Results for Survey Year: 10
##
## Age groups:
##
##   3   1   2   4   5
## 216 346 199 231 182
##
## Age groups percentage:
```

```
## # A tibble: 5 x 3
##   AgeG     pct  pct_se
##   <fct>  <dbl>   <dbl>
## 1 3     0.298  0.0197
## 2 1     0.121  0.00774
## 3 2     0.0785 0.00675
## 4 4     0.280  0.0185
## 5 5     0.222  0.0199
##
## Sex:
##
##   M   F
## 533 641
##
## Sex percentage:
## # A tibble: 2 x 3
##   Sex     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 M     0.486 0.0220
## 2 F     0.514 0.0220
##
## Income tertiles:
##
##   1   2   3
## 338 339 337
##
## Income tertiles percentage:
## # A tibble: 4 x 3
##   eqv     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 1     0.247 0.0204
## 2 2     0.286 0.0230
## 3 3     0.309 0.0207
## 4 <NA>  0.158 0.0197
##
## Meat consumers:
##
##    0    1
##   64 1110
##
## Meat consumers percentage:
## # A tibble: 2 x 3
##   meat_gt_0    pct  pct_se
##       <dbl>  <dbl>   <dbl>
## 1         0 0.0640 0.00956
## 2         1 0.936  0.00956
##
## Results for Survey Year: 11
##
## Age groups:
##
##   3   1   2   4   5
## 208 293 205 187 183
##
```

```
## Age groups percentage:
## # A tibble: 5 x 3
##   AgeG     pct  pct_se
##   <fct>  <dbl>   <dbl>
## 1 3     0.295  0.0215
## 2 1     0.115  0.00740
## 3 2     0.0893 0.00786
## 4 4     0.254  0.0175
## 5 5     0.247  0.0206
##
## Sex:
##
##   M   F
## 493 583
##
## Sex percentage:
## # A tibble: 2 x 3
##   Sex     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 M     0.476 0.0219
## 2 F     0.524 0.0219
##
## Income tertiles:
##
##   1   2   3
## 315 307 312
##
## Income tertiles percentage:
## # A tibble: 4 x 3
##   eqv     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 1     0.262 0.0237
## 2 2     0.273 0.0195
## 3 3     0.301 0.0238
## 4 <NA>  0.164 0.0227
##
## Meat consumers:
##
##    0    1
##   54 1022
##
## Meat consumers percentage:
## # A tibble: 2 x 3
##   meat_gt_0    pct pct_se
##       <dbl>  <dbl>  <dbl>
## 1         0 0.0660 0.0112
## 2         1 0.934  0.0112
##
## Results for Survey Year: 9
##
## Age groups:
##
##   3   1   2   4   5
## 231 361 200 220 204
```

```
##
## Age groups percentage:
## # A tibble: 5 x 3
##   AgeG     pct  pct_se
##   <fct>  <dbl>   <dbl>
## 1 3     0.306  0.0201
## 2 1     0.117  0.00692
## 3 2     0.0806 0.00652
## 4 4     0.251  0.0155
## 5 5     0.245  0.0199
##
## Sex:
##
##   M   F
## 576 640
##
## Sex percentage:
## # A tibble: 2 x 3
##   Sex     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 M     0.526 0.0194
## 2 F     0.474 0.0194
##
## Income tertiles:
##
##   1   2   3
## 360 349 357
##
## Income tertiles percentage:
## # A tibble: 4 x 3
##   eqv     pct pct_se
##   <fct> <dbl>  <dbl>
## 1 1     0.256 0.0220
## 2 2     0.268 0.0205
## 3 3     0.320 0.0232
## 4 <NA>  0.156 0.0246
##
## Meat consumers:
##
##    0    1
##   59 1157
##
## Meat consumers percentage:
## # A tibble: 2 x 3
##   meat_gt_0    pct pct_se
##       <dbl>  <dbl>  <dbl>
## 1         0 0.0710 0.0158
## 2         1 0.929  0.0158
```